Studies in Computational Intelligence 1166

Abdellah Idrissi *Editor*

Modern Artificial Intelligence and Data Science 2024

Tools, Techniques and Systems

Springer

Studies in Computational Intelligence

Volume 1166

Series Editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the world-wide distribution, which enable both wide and rapid dissemination of research output.

Indexed by SCOPUS, DBLP, WTI AG (Switzerland), zbMATH, SCImago.

All books published in the series are submitted for consideration in Web of Science.

Abdellah Idrissi
Editor

Modern Artificial Intelligence and Data Science 2024

Tools, Techniques and Systems

Springer

Editor
Abdellah Idrissi
Department of Computer Science
Mohammed V University in Rabat
Rabat, Morocco

ISSN 1860-949X ISSN 1860-9503 (electronic)
Studies in Computational Intelligence
ISBN 978-3-031-65037-6 ISBN 978-3-031-65038-3 (eBook)
https://doi.org/10.1007/978-3-031-65038-3

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

Artificial Intelligence and Data Science are two of the most important and interesting areas of the current technological landscape. They have become essential for both organizations and individuals. They are not only changing the way we live and work, but they are particularly transforming businesses and industries around the world.

Artificial Intelligence is, without any contest, at the forefront of technological innovation, offering transformative solutions in various fields. It is more focused in simulating the behavior of a normal human being by machines and providing the means to automate and optimize decision-making processes. It also involves the creation of intelligent machines capable of performing tasks that generally require human intelligence and can provide solutions to many practical questions of daily life and if well oriented, guided and better exploited, it can revolutionize very positively the people's lives.

Artificial Intelligence is beginning to have very beneficial applications in our modern society. It can be used everywhere and in all sectors of activity such as education, industry, health, societal challenges, transport, finance, environment, security, telecoms and in all areas to which one can think. It affects all our lives on a daily basis and influences our choice in one way or another.

Thus, there is a whole ecosystem under designing, which aims to provide a new set of products and services. We cannot predict what could happen in the short, medium and long term, but what is certain is that all services will have, without any doubt, an artificial intelligence component. Its success depends, of course, on the availability of large quantities of data. Without a solid foundation and real advances in data science, it is not possible to develop AI systems that can reason, adapt, learn and improve over time.

The field of Data Science, for its part, has seen rapid growth in recent years, and thanks to the advances of AI, this growth has accelerated further. Artificial Intelligence and Data Science are two distinct fields, but they complement each other so well that their combination has become a powerful force for driving innovation and progress across all industries.

Data Science provides the tools and techniques necessary to ensure that the data used to train AI systems is reliable, accurate and free of bias. This is more than

essential to develop reliable AI systems that can make fair and unbiased decisions. It provides the methods, techniques and tools needed to collect, store, clean and analyze data to identify patterns and extract insights.

Data Science thus focuses more on the study of knowledge extraction and data analysis using various scientific methods such as algorithms and processes. It is in fact a field that involves extracting information and knowledge from data. It involves various processes such as data collection, data visualization, data cleaning, data transformation and predictive modeling. Data Science can be used to solve various problems like detecting fraud, predicting customer behavior, recommending products to users and identifying disease outbreaks.

In this context, the integration of AI and Data Science provides a powerful and comprehensive toolset for creating intelligent systems that can learn from data, reason by using rules, adapt to new situations, make decisions autonomously and solving complex real-world problems. This combination has the potential to transform various industries. It has in fact a wide range of applications in different sectors and can lead to significant improvements in efficiency, precision, cost-effectiveness and unlocking new opportunities across a wide range of industries.

Mixing AI and Data Science has numerous real-world applications, spanning industries such as healthcare, environment, finance, education and manufacturing. For example, in healthcare, AI-powered systems can be used to identify potential health issues and analyze medical images, while data science can be used to manage, store, analyze patient data and identify trends and patterns that can help clinical decision-making. In finance, AI can be used to help detecting fraud and improve risk management, while data science can be used to analyze, manage financial data and identify better opportunities for optimization. In manufacturing, AI can be used to improve quality control and optimize production processes, while data science can be used to analyze supply chain data and optimize logistics.

We thus have a unique opportunity to bridge the gap between research and the job market and also to generalize digitalization and create strong links between authorities, governments, civil societies and citizens around the world.

In this book, several chapters explore the alliance between AI and Data Science and illustrate how this marriage can contribute to the construction of intelligent systems. This book consists of five parts, each containing 10 chapters.

In Part I, entitled Artificial Intelligence, Machine Learning and Deep Learning, the reader will discover everything that revolves around Machine Learning and Deep Learning. Machine learning, which is a common component of AI and Data Science, encompasses the development of algorithms and systems capable of learning from data, making predictions and/or decisions based on this data and also improve over time.

Thanks to machine learning, AI algorithms can be trained on large data sets to recognize patterns, do classification, clustering and make predictions, allowing them to perform tasks such as text and voice recognition, object and image classification and natural language processing. This allows organizations to achieve better results, save time and gain insights into their data in real time, thereby improving their decision-making.

Deep Learning is a subset of machine learning. It uses artificial neural networks to simulate the functioning of the human brain and thus allowing making decisions that are more complex.

Part II, composed of ten chapters, is reserved to the theme of computer vision and NLP, which are other fields of AI that have many practical applications. Computer Vision allows using methods and algorithms to manage, analyze and interpret digital images and videos. One application of computer vision is text, facial and fingerprint recognition, which is used for privacy and security purposes. Another application is object recognition, which can be used to identify objects, images and videos. For example, self-driving cars use computer vision to detect pedestrians, obstacles and navigate roads. Computer vision involves the development of powerful algorithms that can manage, analyze and interpret visual data, such as images and videos. As for NLP, it involves the development of tools, methods and algorithms that can understand, manage and process human language.

Part III presents some applications of Artificial Intelligence in Education and eHealth sectors. In this framework, AI combined with Data Science can help enormously in these fields of education and health, as well as in any other field. Education seems to be the most important because everything starts there and from there. In the last 15 years, AI has made significant progress in education. Applications are now widely used by educators and learners, with variations between primary and university environments. In another hand, in the healthcare industry, the combination of Data Science and AI can be used to develop predictive models that can diagnose diseases and recommend treatment plans. The reader will find here ten chapters that deal with these sectors.

Part IV presents applications in the fields of IoT and security in general. Data privacy and security are critical concerns when combining AI and Data Science. This section includes Mapping Algorithms for Enhanced Autonomous Navigation, Enhancing Security in Edge Computing, Design of High-efficiency Power Amplifier using Nonlinear Device Modeling for IoT systems, Discovered Process-Aware IoT Models through Semantic Enrichment, Predicting Credit Risk of SMEs, Advanced Credit Card Fraud Detection, Malware classification in cloud computing using transfer learning and Multi-Agent Reinforcement Learning (MARL) for coordinating autonomous robots.

Finally, Part V includes methods revolving around services and some real-world problems, particularly in the context of automation or decision-making support. In this context, the readers will find in these ten chapters some studies on Optimization Strategies, Ontology Engineering, Data Quality Assessment, Supply Chain Management, Prediction of Railway Infrastructure Defects, Channel Allocation Problem, Wind Energy and Hydraulic Storage Systems and finally a survey about Learning-based Variable Speed Limit control strategies.

We consider that this book presents some important and real interesting advances in the field of Artificial Intelligence combined with Data Science and their applications. It contributes to their emergence, evolution and, particularly, their guidance in the service of the humanity.

We would like to thank all the authors for their interactions, involvements and interesting contributions.

In addition, we would like to warmly thank and sincerely acknowledge the great efforts of the editors, especially Prof. Janusz Kacprzyk, Dr. Thomas Ditzinger, Dr. Sylvia Schneider, Dr. Saranya Sakkarapani and Dr. Michela Castrica for their great help and support and to any person who contribute to promote the Springer Nature Publisher, particularly the Series of Computational Intelligence Studies.

Rabat, Morocco Prof. Abdellah Idrissi

Contents

Artificial Intelligence, Machine Learning and Deep Learning

Quantum Denclue Algorithm (QDA) as a New Clustering Approach Within Quantum Machine Learning 3
Fedoua El Omari, Abdellah Idrissi, Abder Koukam, and Abdeljalil Abbas-Turki

Proposal of Three Algorithms Improving the DENCLUE Algorithm for Data Clustering 23
Khaoula Enaimi and Abdellah Idrissi

Distributed DENCLUE Algorithm Based on Apache Spark 37
Abdellah Idrissi, Khawla Elansari, and Soukaina Elilali

Distributed Evostream Algorithm Based on Apache Spark 51
Abdellah Idrissi, Khawla Elansari, and Mahmoud Lham

Explainable Multi-agent Network for Multi-classification in Small Tabular Data 65
Mehdi Bouskri and Abdellah Idrissi

Agriculture Recommendation System Using Collaborative Filtering 75
Chahrazad Lagrini and Abdellah Idrissi

A Hybrid Ensemble Approach Integrating Machine Learning and Deep Learning with Sentence Embeddings for Webpage Content Classification 85
Kerkri Abdelmounaim and Mohamed Amine Madani

Creating a Customized Dataset for Financial Pattern Recognition in Deep Learning 99
Mustapha El Bakai, Youness Boutyour, and Abdellah Idrissi

Bridging the Gap Between Ontology Engineering and Software Engineering 119
Ouassila Labbani Narsis and Christophe Nicolle

Artificial Intelligence, Computer Vision and NLP

A Re-assessment of Code2Vec 133
Oumaima Bel Moudden, Rym Guibadj, Denis Robilliard, Cyril Fonlupt, Abdeslam Kadrani, and Rachid Benmansour

Evaluating the Use of Feature Extraction and Windowing Using Neural Network in EEG-Based Emotion Recognition 141
Manal Hilali, Abdellah Ezzati, and Said Ben Alla

Arrhythmia Detection in Single-Lead Heartbeat Using ECG Residual Architecture 151
Nadia Berrahou, Hatim Jamali, Abdelmajid El Alami, Abderrahim Mesbah, Rachid El Alami, Hassan Qjidaa, and Aissam Berrahou

Proposed Hybrid Model of Focused Crawler Based on Images Containing Tables 167
Hayat Ouadi, Ilhame El Farissi, and Ilham Slimani

Vehicle Detection in Stereoscopic Images Using Symmetry-Based Approach 179
El Asri Soufiane and Zebbara Khalid

Enhancing Arabic Sentiment Analysis Using AraBERT and Deep Learning Models 189
Abderrahim Ouza, Ali Ouacha, Abdelhamid Rachidi, Mohamed El Ghmary, and Ali Choukri

Morphosyntactic Meaning of Arabic Words 201
Khaireddine Bacha

Semantic Similarity Between Arabic Questions Using Support Vector Machines and Hungarian Method 213
Samira Boudaa, Anass El Haddadi, and Tarik Boudaa

Evaluating Customer Segmentation Efficiency via Sentiment Analysis: An E-Commerce Case Study 223
Lahcen Abidar, Ikram El Asri, Dounia Zaidouni, and Abdeslam En-Nouaary

Enhancing Sentiment Analysis in Moroccan Mixed Script: A Case Study of Perspectives on Distance Learning During the Covid-19 Pandemic 235
Monir Dahbi, Samir Mbarki, and Rachid Saadane

Artificial Intelligence in Education and eHealth

Leveraging Artificial Intelligence (AI)-Enhanced STEM Cognition-Multi-Directionality of Influence 253
Anass Bayaga

Artificial Intelligence and Assessment Generators in Education: A Comprehensive Review 265
Youness Boutyour, Abdellah Idrissi, and Lorna Uden

Personalized Course Recommender System Based on Multiple Approaches: A Comparative Analysis 285
Hajar Majjate, Youssra Bellarhmouch, Adil Jeghal, Ali Yahyaouy, Hamid Tairi, and Khalid Alaoui Zidani

Gamification as a Teaching Strategy for Enhancing Math Problem-Solving Skills in AI: A South African Perspective 295
Janine Olivier, Anass Bayaga, and Greyling Jean

Exploring the Impact of Artificial Intelligence in Education: A Comprehensive Review and Future Directions 307
Said Ouabou and Abdellah Idrissi

A Multi-agent Approach for Intelligent and Cooperative Learning Systems 319
Khaireddine Bacha

AI in Adaptive Learning: Challenges and Opportunities 329
Aicha Er-Rafyg, Hajar Zankadi, and Abdellah Idrissi

Advancements in Artificial Intelligence for Healthcare Systems: Enhancing Efficiency, Quality, and Patient Care 343
Abatal Ahmed, Anass Elachhab, and Elkaim Billah Mohammed

Machine Learning Approach Versus AutoML to Predict the Bioactivity of a Therapeutic Target Related to Cancer 357
Abdellah Idrissi, Khawla Elansari, and Fatima Zahra El Houti

Integrating Artificial Intelligence with Information Systems in Healthcare Supply Chain Management 367
Sabrina Guetibi

Artificial Intelligence in Transport, IoT and Security

Comparative Analysis of Simultaneous Localization and Mapping Algorithms for Enhanced Autonomous Navigation 377
Slama Hammia, Anas Hatim, Abdelilah Haijoub, and Ahmed El Oualkadi

Machine Learning to Predict Railway Infrastructure Defects 391
Khawla Elansari, Abdellah Idrissi, and Hajar Tifernine

Discovered Process-Aware IoT Models Through Semantic Enrichment 407
El Kodssi Iman and Sbai Hanae

Predicting Credit Risk of SMEs in Malaysia: Machine Learning vs Deep Learning 417
Syahida Abdullah and Roshayu Mohamad

Malware Classification in Cloud Computing Using Transfer Learning 429
Meryem EC-Sabery, Adil Ben Abbou, Abdelali Boushaba, Fatiha Mrabti, and Rachid Ben Abbou

An Assessment System for ML-Based XSS Attack Detection Models Between Accuracy Coverage and Data 441
Maryam Et-tolba, Charifa Hanin, and Abdelhamid Belmekki

Integrating Artificial Neural Networks and Support Vector Machines Machine Learning Algorithms for Advanced Credit Card Fraud Detection 453
Oussama Ndama, Ismail Bensassi, and El Mokhtar En-Naimi

Enhancing Security in Edge Computing with RSA and Paillier Encryption Scheme 463
Hamid El Bouabidi, Mohamed EL Ghmary, Salah Eddine Hebabaze, and Mohamed Amnai

Artificial Intelligence in Services and Real Problems

EDNBC: A New Efficient Distributed Naive Bayes Classifier for Vertically Distributed Data 475
Ahmed M. Khedr, Ibrahim Attiya, and Amal Ibrahim Al Ali

Discrete Reptile Search Algorithm-Based Clustering Technique for Flying Ad Hoc Networks 489
P. V. Pravija Raj, Ahmed M. Khedr, and Reham R. Mostafa

Data Quality Assessment of a Utility Company's Geographic Information System 501
Souhaila Akrikez, Mohammed Ammari, and Abdellah Idrissi

ELK Stack Approach with Artificial Intelligence for Logs Collection and Resource Usage Monitoring and Forecasting 515
Khawla Elansari, Abdellah Idrissi, and Kaoutar Moutaouakil

Efficient Wireless Communication in Mobile Edge Computing: Channel Allocation Problem 529
Sara Maftah, Mohamed El Ghmary, and Mohamed Amnai

Optimization Strategies in Mobile Edge Computing Through Intelligent Task Offloading 539
Nouhaila Moussammi, Mohamed El Ghmary, and Abdellah Idrissi

AI for Enhanced Optimal Modeling in Wind Energy and Hydraulic Storage Systems with Lagrangian Insights 555
Abderrahim Ouza, Mohamed El Ghmary, Ali Choukri, and Adil Khazari

A Survey About Learning-Based Variable Speed Limit Control Strategies: RL, DRL and MARL 565
Asmae Rhanizar and Zineb El Akkaoui

Knowledge Management, Decision-Making and Information and Communication Technology: A Systematic Mapping Study 581
Ibtissam Assoufi, Ilhame El Farissi, and Ilham Slimani

What Measurement Scales for Assessing e-reputation? A Systematic Literature Review 593
Mariem Hakim, Catherine Ghosn, and Razane Chroqui

About the Editor

Abdellah Idrissi has graduated with Ph.D. in Artificial Intelligence. He is currently a member of the IPSS team where he leads a research group on Artificial Intelligence and its applications. He is the author of four books and co-author of several publications in international journals and conferences. He is also the co-author of two patents and others are pending. He was a guest editor of five special issues in renowned journals. He is a member of the editorial board of several international journals and a member of the TPC of several international conferences. He is the founder and general chair of two International Conferences, namely, "Modern Artificial Intelligence and Data Science Systems (MAIDSS)" and "Modern Intelligent Systems Concepts (MISC)", and has chaired numerous international conferences and workshops. He has supervised seven doctoral theses, which have been defended with excellence, and many more are in progress. He is the Founder and the Coordinator of the Master on Artificial Intelligence and Data Science. He is a partner of several national and international projects and was particularly a partner of the MOSAIC Project, funded by European Commission (FP7 612076), in which 12 partners representing 12 different countries were participated. He was, in this last project, the Leader in the implementation of the Technology Platform in the Maghreb Region.

Artificial Intelligence, Machine Learning and Deep Learning

Quantum Denclue Algorithm (QDA) as a New Clustering Approach Within Quantum Machine Learning

Fedoua El Omari, Abdellah Idrissi, Abder Koukam, and Abdeljalil Abbas-Turki

Abstract Our current study underscores the central significance of the Quantum kernel within the DENCLUE Clustering Algorithm, providing a novel perspective for understanding data characteristics from a quantum standpoint. While classical kernels depend on conventional distance measures such as Euclidean distance, the quantum kernel employs quantum operators to assess these similarities. Unlike classical kernels, the quantum kernel leverages the unique properties of qubits to precisely quantify data similarities. Our primary objective is to transition classical clustering methods into their quantum counterparts, particularly the various iterations of DENCLUE's Algorithm. The Quantum Kernel offers an advanced approach for exploring non-trivial relationships within data, presenting exciting opportunities for data analysis and processing within a quantum framework, all while striving to significantly enhance the efficiency of data processing across different versions of DENCLUE's clustering algorithms. In this paper, we delve into the concept of a quantum Kernel, highlighting its advancements over classical kernels. This methodology possesses the capability to capture intricate correlations and inherent nonlinear relationships within the data, thereby refining the identification of density attractors of DENCLUE's Algorithms within Quantum Data Sets.

Keywords Clustering · DENCLUE · Quantum machine learning · Quantum kernel · RY gate

F. El Omari (✉) · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team, Faculty of Science of Rabat, Mohammed V University, Rabat, Morocco
e-mail: fedoua.elomari@um5r.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Koukam · A. Abbas-Turki
CIAD, UMR 7533, UTBM, Université Bourgogne Franche-Comté, 90010 Belfort, France
e-mail: abder.koukam@utbm.fr

A. Abbas-Turki
e-mail: abdeljalil.abbas-turki@utbm.fr

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_1

1 Introduction

The synergy between quantum computing and traditional machine learning forms the foundation of quantum machine learning, an emerging interdisciplinary field bridging quantum mechanics and classical machine learning. This domain explores the integration of quantum principles with classical machine learning algorithms to enhance their performance significantly. In this introduction, we delve into the intrinsic relationship between quantum computing and machine learning, discuss the potential advantages of this fusion, and elucidate the fundamental concepts underlying this transformative discipline.

Quantum computing, rooted in the enigmatic realm of quantum mechanics, revolutionizes our understanding of computational power. It transcends the binary paradigm by introducing qubits, the fundamental units of quantum information capable of existing in multiple states simultaneously through superposition. This unique characteristic propels quantum computing beyond the confines of classical computation.

Conversely, classical machine learning aims to impart machines with the ability to learn from data. However, these models are optimized for classical systems, thereby limiting their effectiveness in addressing complex problems.

The amalgamation of quantum computing with machine learning overcomes these limitations by leveraging quantum properties to accelerate computations, handle large datasets, and address particularly complex optimization challenges. This alliance holds great promise for reshaping the future landscape of artificial intelligence and data science.

At the core of this fusion lies the utilization of quantum properties such as superposition and entanglement to enhance the computational capacity of machine learning models. This collaboration has the potential to revolutionize various domains, ranging from artificial intelligence to data analysis.

The potential benefits of quantum machine learning are multifaceted and profound. Primarily, it excels in dramatically accelerating computations by harnessing quantum superposition, thereby reducing training time for complex models and addressing demanding computational tasks. Furthermore, this approach demonstrates remarkable efficiency in handling large-scale datasets, meeting the demands of the big data era. Quantum algorithms are proficient in solving combinatorial optimization challenges, with applications in domains such as logistics planning and drug design. Additionally, quantum computing enhances the security of machine learning systems by strengthening cryptographic techniques, ensuring data confidentiality and integrity.

The integration of quantum computing with the DENCLUE algorithm holds promise for significantly accelerating the analysis of complex data, managing vast datasets, and achieving faster resolution of optimization challenges. This amalgamation heralds exciting prospects for the future of data analysis and the exploration of hidden structures, pushing the boundaries of what was previously conceivable within the realm of traditional machine learning.

To fully grasp quantum machine learning, it is essential to understand key concepts such as superposition, entanglement, and the computation of quantum kernels. Mastery of these concepts lays the groundwork for harnessing the remarkable potential inherent in this field.

In summary, quantum machine learning embodies a transformative domain that leverages quantum computing principles to enhance the performance of machine learning models. It holds immense promise for accelerating computations, managing vast datasets, solving complex problems, and fortifying the security of machine learning systems. As research progresses in this domain, quantum machine learning continues to unveil exciting prospects for the future of artificial intelligence and data science.

The subsequent sections of this article are organized as follows: Sect. 2 provides a review of applied clustering algorithms. (The Sect. 3 explores the integration of quantum machine learning. The Sect. 4.1 focuses on the quantum kernel proposal in DENCLUE. The Sect. 5 encompasses the experimental results, and finally, in Sect. 6, we conclude and discuss future research directions).

2 Clustering Algorithms

Various existing clustering algorithms and methods in the literature are grouped [1, 2] based on many criteria [3], but are generally classified into five major families [4, 5], namely, Partitioning clustering, Hierarchical clustering, Grid-based clustering, Density-based clustering and Model-based clustering.

The partitioning clustering algorithms split data into partitions, then consider each partition as a cluster. The initial clusters are constructed and merged according to certain criteria to get the final results.

The hierarchical clustering algorithms group data into a tree of clusters. This type of clustering is twofold, from top to bottom (divisive) and from bottom to top (agglomerative) [6]. The first one includes all data inside a single cluster before divides it hierarchically into several clusters, until the final clusters are formed. The second one puts every object contained in the database into a cluster, then merges them recursively until the resulted clusters are constructed.

The grid-based clustering algorithms spread the data within a grid. Thus, the algorithm is applied on the grid instead of being applied on the database objects.

The density-based clustering algorithms classifies the objects according to their regions of density. These algorithms are able to come up with clusters of arbitrary shapes and omit noisy objects.

The model-based clustering algorithms are based on the hypothesis that data was generated by underlying probability distributions. This type of clustering is designed to emit a model assumption for each cluster, and then find the best fit of the data to the model.

2.1 K-Means

The K-means algorithm is founded on the minimization of the squared distances between each object in the cluster domain and the cluster center. To do so K-means pursues the following steps [7].
Step 1: The K initial cluster centers are arbitrary chosen.
Step 2: The objects x are distributed among K clusters at the t^{th} iteration as presented in the (1) [7].

$$x \in S_j(t) \quad if \|x - z_j(t)\| < \|x - z_i(t)\| \tag{1}$$

where $i = 1, 2, ..., K$, $j = 1, 2, ..., K$, $i \neq$, with $S_j(t)$ denotes the set of objects whose cluster center is z_j.
Step 3: Based on the results of **Step 2**, the new cluster centers $z_j(t+1)$ are computed, with respect to the minimization of the sum of the squared distance between all points in $S_j(t)$ and the new cluster center. Thus, this cluster center is given in (2) [7].

$$z_j(t+1) = \frac{1}{N_j} \sum_{x \in S_j(t)} x \quad j = 1, 2, ..., K \tag{2}$$

where N_j is considered as number of objects in $S_j(t)$.
Step 4: If $z_j(t+1) = z_j(t)$ for $j = 1, 2, ..., K$, the K-means procedure is terminated. Otherwise go to **Step 2**.

2.2 EM

EM algorithm [8] is designed to estimate the maximum likelihood parameters of a statistical model. It approximates the unknown model parameters with two steps: the E and the M steps as formulated in (3) [5] and (4) [5] respectively:
E-step: Compute the expectation of the complete data log-likelihood.

$$Q(\theta, \theta^T) = E[\log \ p(x^g, x^m|\theta)x^g, \theta^T] \tag{3}$$

M-step: Select a new parameter that maximizes the Q-function.

$$\theta^{t+1} = \arg\max_\theta Q(\theta, \theta^T) \tag{4}$$

2.3 DENCLUE

DENCLUE (DENsity-based CLUstEring) [9] is an algorithm based on density functions and grid structure (a hyper-rectangle and hyper-cubes). A resulted cluster is

considered as connected dense components, which grow in any direction that density leads.

For finding the clusters, DENCLUE has to reach a set of points called density attractors, each one is deemed to be a local maximum of the density function. These maximums are found by using Hill Climbing method. DENCLUE algorithm was developed to cluster large multimedia databases [9], because this kind of data are noisy, and require clustering high-dimensional feature vectors.

DENCLUE is a method based on the influence of points between them, which characterize the influence of a given point in other one in its neighbourhood. There exist many influence functions based on the distance between two points x and y like the Gaussian function derived from [9] and presented in (5).

$$f_{Gauss}(x, y) = \exp^{-\left(\frac{d(x, y)^2}{2\sigma^2}\right)}, \tag{5}$$

where $d(x, y)$ is the Euclidean distance between x and y, and σ represents the radius of the neighbourhood containing x. The total sum of these influence functions defines the density function presented in (6) [9].

$$f_D(x) = \sum_{i=1}^{N} f_{Gauss}(x, x_i), \tag{6}$$

where D corresponds to the set of points on the database, and N is its cardinal.

2.4 DENCLUE 2.0

A second version of DENCLUE called DENCLUE 2.0 was introduced in [10]. In this modified version a new Hill Climbing method for Gaussian kernels was developed. this new version converges exactly to a local maximum, by reducing the original Hill Climbing to a particular instance of Expectation Maximization (EM) algorithm. Consequently the step size are adjusted at no extra costs. The modifications are applied on the gradient ascent approach used by the classical Hill Climbing method as shown in (7) [10].

$$x = \frac{\sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right) x_i}{\sum_{i=1}^{N} K\left(\frac{x - x_i}{\sigma}\right)}, \qquad x^{(l+1)} = \frac{\sum_{i=1}^{N} K\left(\frac{x^{(l)} - x_i}{\sigma}\right) x_i}{\sum_{i=1}^{N} K\left(\frac{x^{(l)} - x_i}{\sigma}\right)}, \tag{7}$$

with K is the Gaussian Kernel presented in (8).

$$K(x) = (2\pi)^{\left(-\frac{d}{2}\right)} \exp\left[-\frac{x^2}{2}\right], \tag{8}$$

where d is the data dimension. Equation (7) is reduced in the two EM steps as shown in (9) and (10) [10].

Step E:

$$\theta_i = \frac{1/N.K\left(\frac{x^{(l)} - x_i}{\sigma}\right)}{f_D(x^{(l)})} \tag{9}$$

Step M:

$$x^{(l+1)} = \frac{\sum_{i=1}^{N} \theta_i x_i}{\sum_{i=1}^{N} \theta_i} \tag{10}$$

2.5 DENCLUE-SA

DENCLUE-SA, improved version of the existing DENCLUE algorithm, aims to improve the local maximum search and proved its capacity to reduce the execution time of DENCLUE especially in large data [11]. This algorithm tried to adapt the Simulated Annealing algorithm and introduced it instead of the Hill Climbing.

2.6 DENCLUE-GA

DENCLUE-GA is another improvement of DENCLUE, which exceeds in the most case the DENCLUE and DENCLUE-SA run time, especially when the size of data is large [11]. In this version, DENCLUE algorithm has been modified by alterning the Hill Climbing step by an adapted Genetic Algorithm presented in [12].

2.7 DENCLUE-IM

The research of density attractors becomes more difficult on large data, that is why DENCLUE-IM have been developed [13, 14]. The algorithm modifies the step of computing density attractors that are found by the Hill Climbing algorithm. This step, based on gradient calculations, is done for every point to discover its own density attractor. DENCLUE-IM considers a representative of all the points contained in a hyper-cube instead of the calculations made for each point from the dataset. The representative of the hyper-cube, denoted x_{Hcube}, is considered as the point having the

highest density in his hyper-cube as shown in (11). Thus each hyper-cube represented by its own x_{Hcube} is considered as an initial cluster. The formed initial clusters are unified on the circumstance that there exist a path between their representatives.

$$\forall x \in C_p \quad f_D(x) \leq f_D(x_{Hcube}), \tag{11}$$

where C_p denote a given populated hyper-cube in the constructed hyper-rectangle.

3 Quantum Machine Learning

Quantum machine learning represents an intriguing convergence between two revolutionary domains: quantum computing and traditional machine learning. This emerging discipline explores how the captivating principles of quantum mechanics can significantly enhance the performance of classical machine learning algorithms. In this introduction, we will delve into the relationship between quantum computing and machine learning, examine the potential advantages of this fusion, and dissect the fundamental concepts that underpin this field [15].

Quantum computing, immersed in the mysteries of quantum mechanics, dramatically reshapes our understanding of computational power. It reinvents the concept of bits by introducing qubits, the fundamental building blocks of quantum information capable of simultaneously occupying multiple states through superposition. This unique characteristic propels quantum computing far beyond the capabilities of classical computing.

On the other hand, machine learning strives to empower machines with the ability to learn from data. However, traditional machine learning models are optimized for classical computers, limiting their effectiveness in tackling complex problems.

When quantum computing joins forces with machine learning, it pushes these boundaries by harnessing quantum properties to accelerate calculations, manage vast volumes of data, and solve particularly challenging optimization problems. This promising convergence opens new horizons for the future of artificial intelligence and data science.

The synergy between quantum computing and machine learning hinges on the captivating idea of utilizing quantum properties such as superposition and entanglement to augment the computational capacity of machine learning models. This alliance holds revolutionary potential across various domains, ranging from artificial intelligence research to data analysis.

The potential benefits of quantum machine learning are manifold and significant. Firstly, it stands out for its ability to dramatically accelerate computations by harnessing quantum superposition, resulting in a remarkable reduction in the time required for training complex models and solving demanding computational problems. Furthermore, this approach proves highly efficient in managing massive data volumes, addressing the imperatives of the big data era. Quantum algorithms also excel in solving combinatorial optimization problems, including the search for optimal solutions

in vast solution spaces, with potential applications in areas such as logistics planning and drug design. Finally, quantum computing bolsters the security of machine learning through enhanced cryptography techniques, ensuring data confidentiality and integrity. These advantages collectively position quantum machine learning as a promising discipline for the future of artificial intelligence and data science.

To fully grasp quantum machine learning, it is essential to master several key concepts, including superposition, entanglement, quantum gates, and specific algorithms such as the Grover algorithm and the Deutsch-Jozsa algorithm.

In summary, quantum machine learning represents a promising discipline that leverages the principles of quantum computing to elevate the performance of machine learning models. It offers immense potential to expedite computations, handle massive data, solve complex problems, and enhance the security of machine learning systems. As research in this field advances rapidly, quantum machine learning continues to unveil exciting new prospects for the future of artificial intelligence and data science.

4 Proposed Algorithm: QDENCLUE

The evolution of Quantum Machine Learning targets a specific challenge in exploring density attractors within extensive datasets, particularly highlighted within DENCLUE's framework. This challenge poses a significant concern in quantum data analysis, often demanding substantial computation time with traditional methods. Integrating the quantum kernel offers a promising solution by leveraging quantum properties like superposition and entanglement. This enhances computational capability, enabling a faster and more precise exploration of density attractors. Employing the quantum kernel improves the speed and accuracy of identifying density attractors, marking a notable advancement in quantum data analysis through DENCLUE.

4.1 Integration of Quantum Kernel Mechanism: Enhancing Density Attractor Identification in DENCLUE

The primary goal of DENCLUE is to accurately identify density attractors within extensive datasets, a crucial task for various applications demanding the detection of significant clusters within complex structures. However, this mission can be particularly demanding, especially when dealing with large-scale databases.

The integration of the quantum kernel mechanism, a key element of the method, plays a central and essential role in our context. This quantum mechanism is characterized by its ability to assess data point similarities by leveraging quantum properties. Operating on principles such as superposition and entanglement, it offers a distinctive approach to evaluate data relationships within extensive datasets.

The general mathematical representation of the quantum kernel might resemble the following formula:

$$K_Q(x, y) = \langle \Phi_x | U^\dagger U | \Phi_y \rangle \tag{12}$$

where:

$K_Q(x, y)$ signifies the quantum kernel value between data points x and y.

Φ_x and Φ_y stand for quantum representations of the data, prepared from the features of points x and y.

U performs a quantum transformation on these representations, with $U^\dagger$ representing the conjugate inverse operator.

Consider the impact of this convergence in the search for density attractors. In a domain where precision and the prompt identification of key points are critical, the integration of the quantum kernel holds significant potential. By employing quantum properties to evaluate data point similarities, the quantum kernel offers a new perspective, potentially accelerating the identification of these attractors within extensive datasets [15].

At the core of this proposal lies the integration of the quantum kernel mechanism within the DENCLUE framework, a classical algorithm for density attractor search. The quantum kernel, leveraging the unique characteristics of quantum mechanics [16], presents a distinct approach to evaluate data point similarities. When applied to DENCLUE, the quantum kernel becomes a crucial element for quickly and precisely identifying density attractors, enhancing the accuracy and expediting the analysis of complex datasets. This intelligent integration opens new avenues to explore complex clusters while accelerating the search process and improving the precision of detecting key points within dense and unstructured datasets.

4.2 Revolutionizing Data Density Calculation: The Integration of Quantum Kernel in DENCLUE Algorithm

The integration of the quantum kernel into the DENCLUE algorithm fundamentally revolutionizes the calculation of data density. This concept relies on harnessing principles of quantum computation, notably through employing the RY gate (Rotation Y) in a quantum circuit [17]. The RY gate operation rotates around the Y-axis of a qubit, generating a quantum representation of the similarity between two data points. This quantum representation is determined by an angle of rotation, θ, computed based on the distance between data points and the parameter h. The mathematical formula of the RY gate is expressed as:

$$RY(\theta) = \begin{bmatrix} \cos(\frac{\theta}{2}) & -\sin(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) & \cos(\frac{\theta}{2}) \end{bmatrix} \tag{13}$$

In the context of the DENCLUE quantum kernel, this rotation angle θ is pivotal as it defines the quantum similarity representation between these points. The method for calculating quantum similarity is defined by the relationship between point distances and the bandwidth parameter h, providing an essential measure to estimate data density in a multidimensional space.

This quantum similarity, determined through the RY gate, is utilized to estimate the density of each data point. Higher quantum similarity results in higher density, facilitating the identification of density-based clusters. This groundbreaking integration of the quantum kernel into DENCLUE paves the way for more precise and efficient results in detecting complex clusters within multidimensional spaces. This innovative mechanism represents a significant advancement in data analysis and the discovery of latent structures.

5 Experimental Results

5.1 Data Description

The "OnlineRetail" dataset originates from the University of California, Irvine (UCI), a renowned institution for its machine learning and data science resources. This dataset provides insights into the transactions of an online retail company, encompassing crucial information regarding customer purchases, sold products, and other transaction-related details. Three columns are particularly relevant for our analysis:

- CustomerID (Unique Customer Identifier): This column assigns a unique identifier to each customer of the company, ensuring a clear distinction among customers.
- StockCode (Product Code): Every product sold by the company is associated with a unique code in this column, facilitating precise product identification.
- Quantity: The "Quantity" column indicates how many units of each product were purchased by a customer during a specific transaction. It reflects the quantity of products included in each purchase.

Our objective is to leverage these three columns, namely "CustomerID," "StockCode," and "Quantity," to perform clustering based on customer purchasing behavior. This approach involves grouping customers based on their buying habits, taking into account the quantity of products purchased (Quantity), and identifying the specific products (StockCode) included in their purchases. This clustering will enable us to define customer segments that share similarities in their purchasing behaviors. Subsequently, this information can be used for marketing personalization, product recommendations, and other sales-related analyses, contributing to a better understanding of customer preferences and needs.

5.2 Validity Metrics

Several validity metrics [5, 18, 19] are used in order to assess the performance of the clustering results. These metrics are splitted into two types of validation measures: internal and external.

In the internal type, the computations are made without any additional information such as class labels. Therefore, the internal validity metrics remain the unique option for cluster validation when there is no external information available. The internal measures include the Dunn Index [20], the Davies-Bouldin Index [21] and the Compactness Index [5].

The Dunn Index (DI), evaluates the separation degree between individuals of the same cluster (intra-cluster similarity). A high value indicates better clustering.

The Davies-Bouldin Index (DBI) behaves as DI, besides, it evaluates the separation degree between clusters (inter-cluster dissimilarity), the smallest value indicates better clustering.

The Compactness Index (CP) calculates the mean distance between pairs including in the same cluster. Then, it remake the same computation between all the clusters. Each member in the cluster has to be too close of others, thus the lowest value indicates the bestest clustering quality.

When true cluster labels are unavailable, calculating evaluation metrics such as accuracy becomes impossible. However, entropy can be utilized to assess the consistency of clusters formed by DENCLUE. Entropy measures the disorder within clusters, where lower entropy indicates greater similarity among their members. This measurement can be derived from the data point distribution within each cluster. A lower overall entropy suggests a better clustering structure.

Cluster evaluation remains a complex domain, and entropy offers insight into the internal consistency of clusters. Nonetheless, for a comprehensive evaluation, additional measures and exploratory analyses might be required to assess the quality of the obtained clusters

5.3 Results

5.3.1 Interpretation of Clustering Results

Interpreting the results for each measurement across different variants of the Denclue algorithm:

- DBI (Davies-Bouldin Index): DENCLUE 2.0 and QDENCLUE have the lowest DBI values (1.49 and 1.47, respectively), indicating better cluster separation and compactness compared to the other variants (see Fig. 1).

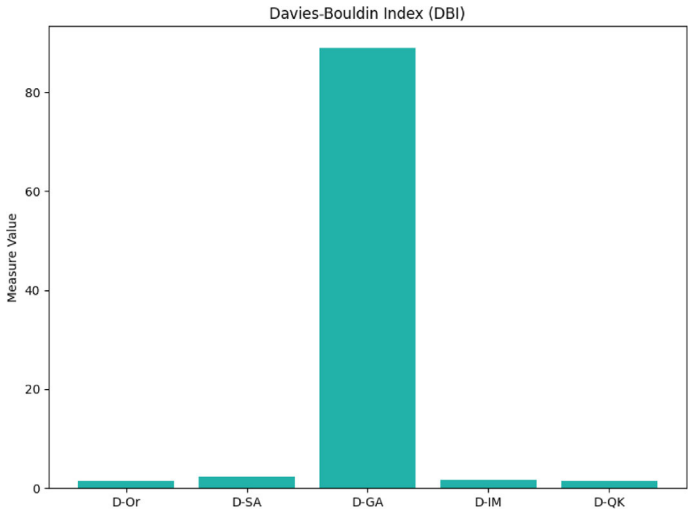


Fig. 1 DBI versus algorithm variants

- CP (Compactness): QDENCLUE exhibits the lowest compactness score (12.9), suggesting better overall compactness of clusters compared to the other variants (see Fig. 2).

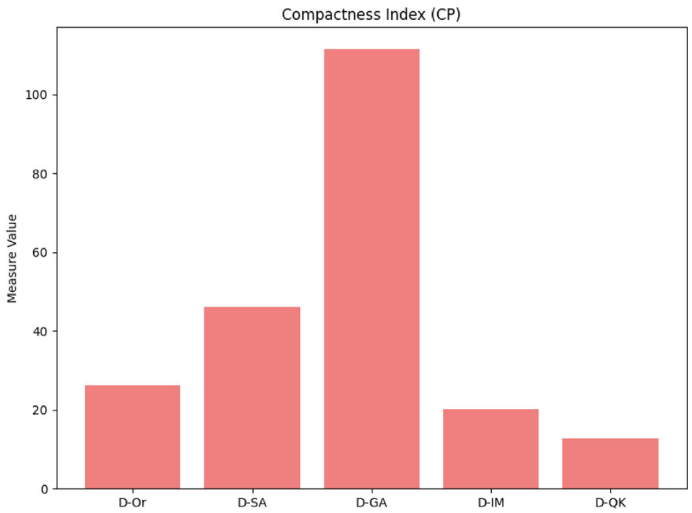


Fig. 2 CP versus algorithm variants

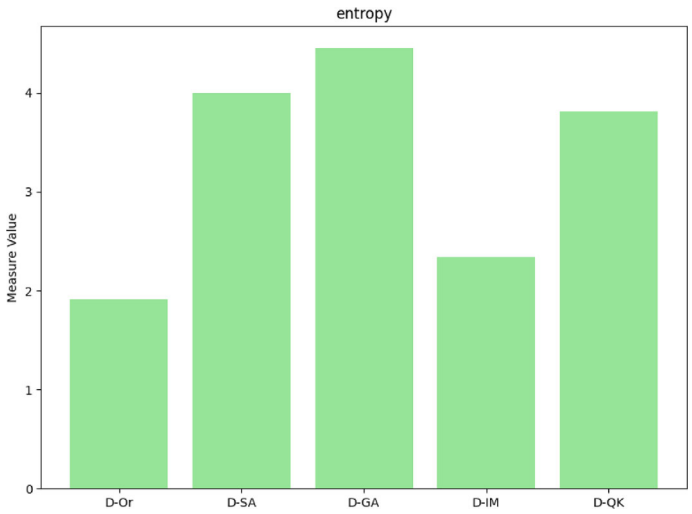


Fig. 3 Entropy versus algorithm variants

- Entropy: DENCLUE 2.0 demonstrates the lowest entropy (1.91), signifying better-defined and more homogeneous clusters compared to other variants (see Fig. 3).
- Silhouette Score: DENCLUE 2.0 and QDENCLUE display higher silhouette scores (0.007 and 0.02, respectively), suggesting relatively better-defined clusters compared to other variants (see Fig. 4).

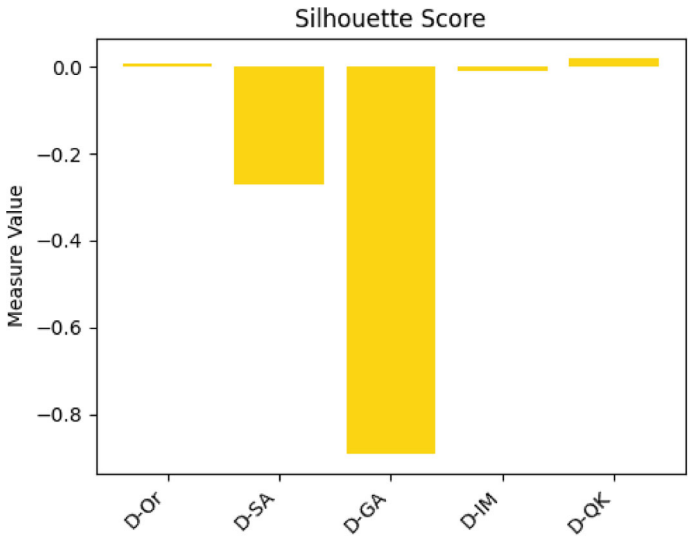


Fig. 4 Silhouette versus algorithm variants

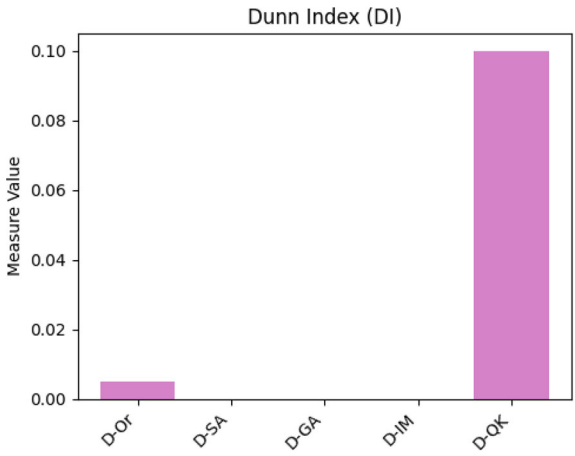


Fig. 5 DI versus algorithm variants

- DI (Dunn Index): QDENCLUE shows a slightly higher value (0.1), indicating better cluster separation compared to other variants, although the differences are minimal (see Fig. 5).

In summary, considering each specific measurement:

- **Cluster separation (DBI)**: DENCLUE 2.0 and QDENCLUE perform the best.
- **Cluster compactness (CP)**: QDENCLUE demonstrates the best compactness.
- **Cluster definition and homogeneity (Entropy)**: DENCLUE 2.0 achieves the best performance.
- **Cluster definition by silhouette method (Silhouette Score)**: DENCLUE 2.0 and QDENCLUE have the highest scores.
- **Cluster separation (DI)**: QDENCLUE exhibits a slight improvement compared to other variants.

Overall, the results suggest that DENCLUE 2.0 and QDenclue stand out in terms of cluster separation, compactness, and definition compared to the other variants.

5.3.2 Customer Purchase Behavior-Based Clustering

Clustering based on the "CustomerID," "StockCode," and "Quantity" columns generated a total of 540 clusters. By examining the purchase frequency categories within these clusters, we can draw several conclusions:

- Variation in purchase frequency: The clusters show significant variation in customer purchase frequency, indicating that customers are grouped based on their buying habits.

- Customer segmentation: The clusters represent different customer segments with similar purchasing behaviors.
- Marketing targeting opportunity: Businesses can tailor their marketing strategies based on the characteristics of each cluster.
- Inventory optimization: Companies can better manage their inventory by understanding which products are popular in each cluster.
- Ongoing evaluation: It is recommended to regularly reassess the clusters to maintain their relevance (see Tables 1 and 2).

Table 1 Denclue-metrics

Measure	D.O	D-SA	D-GA	D-IM	D-QK
DBI	1.49	2.36	88.87	1.57	1.47
CP	26.24	46.18	111.54	20.25	12.9
Entropy	1.91	4	4.45	2.34	3.81
Silhouette	0.007	–0.27	–0.89	–0.01	0.02
DI	0.005	0.0	0.0	0.0	0.1

Table 2 Results of customer purchase behavior-based clustering

Cluster	Purchase frequency	Number of customers
1	High	25
2	Moderate	50
3	Low	75
4	High	30
5	Low	60
...	...	...
540	Low	45

5.3.3 Execution Time

See Table 3, Figs. 6 and 7.

Table 3 Execution times of DENCLUE algorithms

Algorithm	D-Or	D-SA	D-GA	D-IM	D-QK
Time (s)	380	907	619	26 s	180

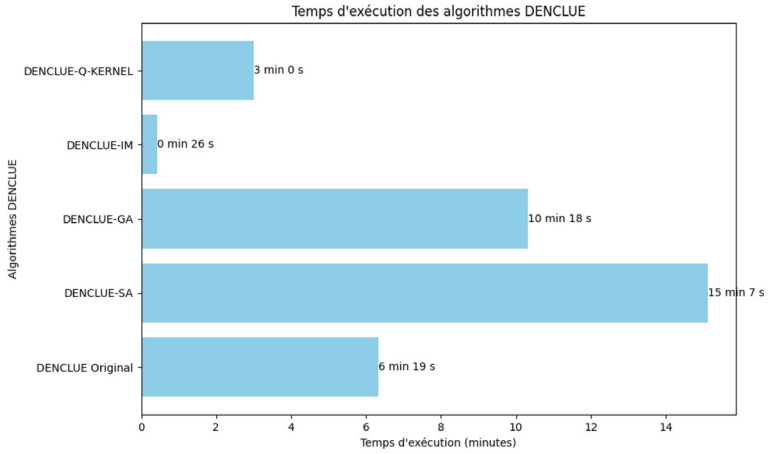


Fig. 6 Execution times of DENCLUE algorithms

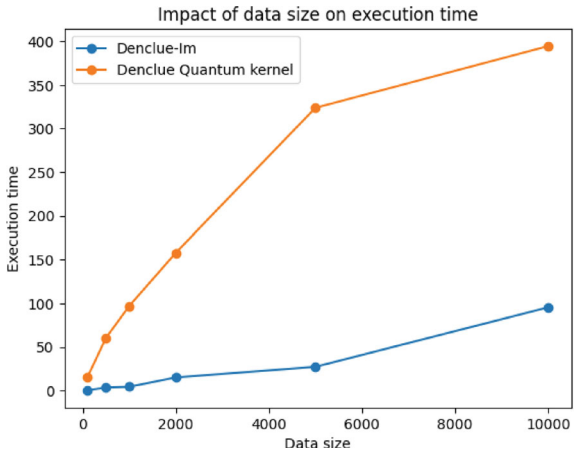


Fig. 7 Execution time comparison: Denclue-lm vs. Denclue quantum kernel

6 Conclusion

The analysis of various measures indicates that among the studied DENCLUE variants, DENCLUE-IM and QDENCLUE show the most promising results.

DENCLUE-IM: Although slightly slower in processing, DENCLUE-IM exhibits the lowest entropy among the variants. This suggests higher homogeneity within clusters, which is beneficial for analyses emphasizing internal coherence of the groups. QDENCLUE: This variant demonstrates solid performance, displaying relatively lower values in most measures, with an execution time of only 3 min. Additionally, the silhouette score shows slightly better cluster quality compared to other variants.

Conversely, DENCLUE-SA shows inferior performance, with less satisfactory results in terms of cluster compactness (DBI and CP), potentially restricting its applicability in scenarios requiring more compact and coherent clusters.

Overall, the performance varies based on specific analysis priorities. The choice of a variant strongly relies on the requirements for cluster compactness, coherence, and execution efficiency.

Ultimately, our study highlights the importance of choosing the clustering method based on the specific objectives of the analysis. Data analysts should carefully assess priorities regarding compactness, coherence, and execution time to select the method best suited to their use case.

Furthermore, customer purchase behavior-based clustering, using the "CustomerID," "StockCode," and "Quantity" columns, provides an effective approach to understanding customer habits. This segmentation can be leveraged for more targeted marketing campaigns, optimal inventory management, and personalized customer offers. However, it is crucial to regularly maintain and update these clusters to track changes in customer behavior and ensure the relevance of business strategies. Furthermore, numerous algorithms could be tested in this area including those exposed in [22–38] which could be adapted to more contribute to the concept of Quantum Machine Learning.

References

1. Berkhin, P.: A survey of clustering data mining techniques. In: Grouping Multidimensional Data, pp. 25–71. Springer (2006)
2. Xu, R., Wunsch, D., et al.: Survey of clustering algorithms. IEEE Trans. Neural Netw. **16**(3), 645–678 (2005)
3. Jain, A.K., Topchy, A., Law, M.H., Buhmann, J.M.: Landscape of clustering algorithms. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004 (ICPR'04), vol. 1, pp. 260–263. IEEE (2004)
4. Shah, G.H., Bhensdadia, C., Ganatra, A.P.: An empirical evaluation of density-based clustering techniques. Int. J. Soft Comput. Eng. (IJSCE) (2012). ISSN: 2231–2307
5. Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A.Y., Foufou, S., Bouras, A.: A survey of clustering algorithms for big data: taxonomy and empirical analysis. IEEE Trans. Emerg. Top. Comput. **2**(3), 267–279 (2014)

6. Ding, C., He, X.: Cluster merging and splitting in hierarchical clustering algorithms. In: IEEE International Conference on Data Mining (ICDM'02), pp. 139–146. IEEE (2002)
7. Albayrak, S.: Unsupervised clustering methods for medical data: an application to thyroid gland data. In: Artificial Neural Networks and Neural Information Processing-ICANN/ICONIP 2003, pp. 695–701. Springer (2003)
8. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. J. R. Stat. Soc. Ser. B, 1–38 (1977)
9. Hinneburg, A., Keim, D.A.: An efficient approach to clustering in large multimedia databases with noise. In: KDD, vol. 98, pp. 58–65 (1998)
10. Hinneburg, A., Gabriel, H.H.: Denclue 2.0: Fast clustering based on kernel density estimation. In: Advances in Intelligent Data Analysis VII, pp. 70–80. Springer (2007)
11. Idrissi, A., Rehioui, H., Laghrissi, A., Retal, S.: An improved DENCLUE algorithm for data clustering. In: IEEE 2015 International Conference on Information and Communication Technology and Accessibility (ICTA'15). IEEE (2015)
12. Idrissi, A., Zegrari, F.: A new approach for a better load balancing and a better distribution of resources in cloud computing. Int. J. Adv. Comput. Sci. Appl. **6**(10) (2015)
13. Rehioui, H., Idrissi, A., Abourezq, M., Zegrari, F.: DENCLUE-IM: a new approach for big data clustering. In: The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016), May 23–26, 2016, Madrid, Spain, pp. 560–567 (2016)
14. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017). https://doi.org/10.1504/IJBIDM.2017.10008309
15. Pérez-Salinas, A., Cervera-Lierta, A., Gil-Fuster, E., Latorre, J.I.: Data re-uploading for a universal quantum classifier. Quantum **4**, 226 (2020). https://doi.org/10.22331/q-2020-02-06-226
16. Magano, D., Buffoni, L., Omar, Y.: Quantum density peak clustering. Quantum Mach. Intell. **5**(9), 1–2 (2023)
17. Poggiali, A., Berti, A., Bernasconi, A., Corso, G.M.D., Guidotti, R.: Quantum clustering with k-means: a hybrid approach (2022). arXiv:2212.06691 [quant-ph]
18. Cai, X., Nie, F., Huang, H.: Multi-view k-means clustering on big data. In: Proceedings of 23rd International Joint Conference on Artificial Intelligence, pp. 2598–2604. AAAI Press (2013)
19. Liu, Y., Li, Z., Xiong, H., Gao, X., Wu, J., Wu, S.: Understanding and enhancement of internal clustering validation measures. IEEE Trans. Cybern. **43**(3), 982–994 (2013)
20. Dunn, J.C.: Well-separated clusters and optimal fuzzy partitions. J. Cybern. **4**(1), 95–104 (1974)
21. Davies, D.L., Bouldin, D.W.: A cluster separation measure. IEEE Trans. Pattern Anal. Mach. Intell. **2**, 224–227 (1979)
22. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular ad-hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
23. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv:1307.5910
24. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
25. Idrissi, A., Li, C.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
26. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
27. Idrissi, A., Yakine., F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
28. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
29. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)

30. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Arch. **9**(2–3), 136–148 (2020)
31. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless ad hoc networks using the skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of Things and Cloud Computing (2016)
32. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
33. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol., 5567–5584 (2023)
34. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
35. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
36. ElHandri, K., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
37. Elhandri, K., Idrissi, A.: Comparative study of Top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
38. Elhandri, K., Idrissi, A.: Parallelization of top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–48862021 (2020)

Proposal of Three Algorithms Improving the DENCLUE Algorithm for Data Clustering

Khaoula Enaimi and Abdellah Idrissi

Abstract One of the most widely used techniques in Machine Learning is clustering. It is an approach for processing unlabeled data to identify homogeneous groups. This approach requires the construction of efficient models to develop high-performance algorithms. The use of clustering methods allows data to be organized into more manageable categories, thus simplifying the analysis, understanding and interpretation of the information contained in this complex data. This approach has prompted the development of new algorithm alternatives aimed at improving data clustering. In this perspective, various clustering techniques have been put forward, notably the DENCLUE algorithm, which uses the density of the data to detect clusters. It relies on the Hill Climbing algorithm to provide support in the crucial phase of class reconstruction. The aim of this paper is to evaluate the performance of this algorithm using strategies based on the three Hill Climbing variants (Simple Hill Climbing, Steepest Ascent Hill Climbing and Stochastic Hill Climbing) on the different data sets.

Keywords Machine Learning (ML) · Clustering · DENCLUE · Hill climbing · Simple hill climbing · Steepest ascent hill climbing · Stochastic hill climbing

1 Introduction

Nowadays, the world is currently facing a huge explosion of massive data. Astronomical quantities of information are being generated every day, in a large variety of fields and multiple formats. This variety of data requires a particular approach to organizing it efficiently. In light of this challenge, intelligent data processing is becoming a necessity. This is the cornerstone for extracting valuable insights and

K. Enaimi (✉) · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team, FSR, Mohammed V University, Rabat, Morocco
e-mail: khaoula.enaimi@um5r.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_2

taking appropriate decisions. At the heart of this processing stand Machine Learning methods, particularly clustering, i.e. grouping similar data into distinct sets. Clustering holds a crucial role in the field of Machine Learning, providing an essential method for dividing data into coherent clusters based on their similarities. The main partitioning step is based on a variety of models, each associated with a different set of algorithms [1].

The main purpose of this study is to focus on the powerful category of density-based clustering methods. This orientation is motivated by the ability of these methods to detect clusters of various forms while reducing undesirable interference. In this context, the DENCLUE algorithm has been adopted. It attributes a density function to each data point, in order to identify high-density modes that represent cluster centers. This approach is characterized by its efficiency in handling high-dimensional data, founded on a robust mathematical background. The model proceeds through two steps: the first involves structuring the data in a search space represented within an hyper rectangle, followed by the identification and combination of populated cubes to create a new map. In the second phase of the clustering process, the map is used to detect a set of points called density attractors. These points are identified using the Hill Climbing algorithm. Finally, these points are linked according to their proximity in order to assign them to the respective clusters [2].

Meanwhile, the DENCLUE method has been improved in a multitude of significant application domains, in areas including medicine [3], sentiment analysis for Twitter [4], Big Data [5], and also for the exploration and optimization of cloud services [6]. Nonetheless, a major shortcoming of DENCLUE is related to the Hill Climbing algorithm, which may not converge to an optimal local maximum, leading to significantly slower execution times.

To overcome this situation, several improvements have been introduced. Among these are DENCLUE-GA, based on the genetic algorithm, and DENCLUE-SA, based on simulated annealing [7]. A further iteration of DENCLUE, baptised DENCLUE 2.0 and developed by Hinneburg and Gabriel [2], features a faster Hill Climbing method for locating cluster centers. This method of Hill Climbing automatically adjusts the step size, speeding up the clustering procedure without additional cost. Furthermore, an alternative version called DENCLUE-IM, developed by Rehioui and Idrissi [5], takes the step based on the Hill Climbing algorithm and performs gradient calculations for each point to determine its density attractor.

The present work aims to evaluate the performance of the different variations of the DENCLUE algorithm, exploiting the three versions of the Hill Climbing algorithm. Ultimately, to determine which variant offers the optimum clustering quality, using metrics for both internal and external performance evaluation. This paper is organized in the following structure: Sect. 2 is devoted to a description of the DENCLUE algorithm and its variants. Section 3 provides an overview of DENCLUE implementations with the three Hill Climbing versions. Section 3 presents experimental results achieved with the above-mentioned implementations. Finally, Sect. 5 provides a conclusion discussing future perspectives.

2 Exploration of the DENCLUE Algorithm and Its Iterations

2.1 DENCLUE

DENCLUE offers an approach based on the concepts of density and connectivity. This algorithm identifies a cluster as a connected, dense element, capable of expanding in any direction as indicated by its density. It is particularly impressive in its ability to identify clusters within large data sets, even in the presence of substantial noise. An essential feature of DENCLUE is its cluster model, which is driven by the position of a local maximum in the estimated density function.

The task of DENCLUE involves the detection of clusters, by locating sets of points called "density attractors". Attractors are local maxima in the density function of the data, and the points attracted are those that create a connection with these attractors. To achieve this task, DENCLUE deploys a hill climbing algorithm that seeks to find these vertices. Although the algorithm originally designed for the analysis of large multimedia databases, DENCLUE stands out for its ability to handle the noise present in these databases, arising from the high dimensions of the feature vectors.

The basis of DENCLUE is the evaluation of interactions between data points, to quantify the impact of each point on the others in its environment. This approach explores various influence functions, which factor in the distance between x and y points. Among these functions, the Gaussian function, plays a crucial role, and is formulated as follows (1):

$$f_{Gauss}(x, y) = \exp\left(-\frac{d(x, y)^2}{2\sigma^2}\right) \tag{1}$$

where dist(x, y) represents the Euclidean distance between x and y.

Furthermore, σ, representing the radius of the neighborhood containing point x, performs a crucial role in the calculations. Indeed, it is the global sum of the influence functions that defines the size of the neighborhood. This global sum is the basis for defining the density function, as shown below (2):

$$f_D(x) = \sum_{i=1}^{N} f_{Gauss}(x, x_i) \tag{2}$$

where D is the set of points in the database and N is its cardinal.

2.2 DENCLUE 2.0

In 2007, Hinneburg and Gabriel launched a new version of DENCLUE designated DENCLUE 2.0 [2]. This new iteration is based on a new Hill Climbing approach applied to Gaussian kernels, which avoids the need for further steps in the search for the local maximum. This enhancement guarantees accurate convergence to a local maximum by reducing this process to a special case of the expectation-maximization (EM) algorithm.

A major reduction is carried out by DENCLUE 2.0's radical overhaul of the Hill Climbing approach, a significant improvement on the classic gradient climbing method. The new formula is presented as follows (3 and 4):

$$x = \frac{\sum_{i=1}^{N} K\left(\frac{x-x_i}{\sigma}\right) x_i}{\sum_{i=1}^{N} K\left(\frac{x-x_i}{\sigma}\right)} \tag{3}$$

$$x^{(l+1)} = \frac{\sum_{i=1}^{N} K\left(\frac{x^{(l)}-x_i}{\sigma}\right) x_i}{\sum_{i=1}^{N} K\left(\frac{x^{(l)}-x_i}{\sigma}\right)} \tag{4}$$

with K is the Gaussian Kernel presented as follows: (5), where d is the data dimension.

$$K(x) = (2\pi)^{-d/2} \exp\left[-\frac{x^2}{2}\right] \tag{5}$$

2.3 DENCLUE-SA and DENCLUE-GA

- DENCLUE-SA is an enhanced version of the DENCLUE algorithm dedicated to reducing execution time, especially for large data sets, by replacing the Hill Climbing algorithm with another algorithm called Simulated Annealing (SA). The Simulated Annealing is a meta-heuristic approach inspired from the field of metallurgy, used to search for a state of minimum energy corresponding to a stable structure, avoiding the trappings associated with local minima.
- DENCLUE-GA is a further improvement of the DENCLUE algorithm, designed to reduce execution time, in particular for processing big amounts of data. This enhancement is achieved by incorporating an adapted genetic algorithm to replace the Hill Climbing step of DENCLUE. Genetic algorithms represent a type of optimization meta-heuristic inspired by evolutionary theory. It is applied to a population of individuals, attempting to obtain the best possible solution using selection, crossover and mutation operators. Using an objective function, the genetic algorithm estimates the quality of solutions, then iterates to produce new generations by refreshing the existing population [7].

2.4 DENCLUE-IM

Another innovation of the DENCLUE algorithm is DENCLUE-IM, Ref. [5] which is designed to modify the step using the Hill Climbing algorithm. Such a modification is intended to avoid gradient calculations for each data point, which can be time-consuming, especially with large databases. As an alternative, DENCLUE-IM provides an efficient approach to identifying an equivalent of the density attractor by identifying all the points contained in a hyper-cube, which simplifies the calculation process.

3 Alternative Approaches of DENCLUE with Hill Climbing

3.1 DENCLUE with Simple Hill Climbing

The simple Hill Climbing algorithm constitutes the fundamental version of the Hill Climbing local search algorithm. It consists in exploring the neighbors of an initial solution in order to select the first neighbor with the best value. This is the basic version used in the DENCLUE 2.0 Algorithm 1.

3.2 DENCLUE with Steepest Ascent Hill Climbing

Steepest Ascent Hill Climbing is another variant of the Hill Climbing algorithm, which examines all possible neighbors of an initial solution and selects the one with the greatest improvement in terms of value. It works with DENCLUE according to the steps described in the following Algorithm 2.

3.3 DENCLUE with Stochastic Hill Climbing

Stochastic escalation is another derivative of escalation, which first selects a solution at random, then explores its neighbors. If a neighbor improves on the current solution, it replaces it. If not, the current solution is retained. This method is slower than the other versions, and it is implemented in the basic DENCLUE 1.0 version as described in the following Algorithm 3.

Algorithm 1 $DENCLUE - SIHC Algorithm$

```
inputDataSet: input data set
h: density parameter
eps: stop threshold
function DENCLUE with SIHC
clusters ← []
labels ←array of zeros of length equal to the size of inputDataSet
for i, dataPoint in enumerate(inputDataSet) do
    currentCluster ← SIHC(dataPoint)
    if labels[i] == 0 : then
        newCluster ← [i]
        for j in range(i + 1, len(inputDataSet)) : do
            if labels[j] == 0 and distance(SIHC(inputDataSet[j]), SIHC(dataPoint))
    < h : then
                newCluster.append(j)
                labels[j] = len(clusters) + 1
            end if
            clusters.append(newCluster)
        end for
        return labels, clusters
    end if
end for
end function
function SIHC(point)
while True do
    newPoint = None
    bestDensity = −inf
    for neighbor in neighbors(point) : do
        neighborDensity = density(neighbor)
        if neighborDensity > bestDensity : then
            newPoint = neighbor
            bestDensity = neighborDensity
        end if
        if bestDensity − density(point) < eps : then
            return newPoint
        end if
    end for
    point = newPoint
end while
end function
function neighbors(point)
return [point + h ∗ direction for direction in directions] + [point − h ∗ direction for
direction in directions]
end function
function density(point)
return sum(exp(−(norm(point − xi)/h) ∗ ∗2) for xi in inputDataSet)
end function
Exit
```

Algorithm 2 $DENCLUE - SAHC Algorithm$

```
inputDataSet: input data set
h: density parameter
eps: stop threshold
function DENCLUE with SAHC
clusters ← []
labels ←array of zeros of length equal to the size of inputDataSet
for i, dataPoint in enumerate(inputDataSet) do
    currentCluster ← SAHC(dataPoint)
    if labels[i] == 0 : then
        newCluster ← [i]
        for j in range(i + 1, len(inputDataSet)) : do
            if labels[j] == 0 and distance(SAHC(inputDataSet[j]), SAHC(dataPoint))
    < h : then
                newCluster.append(j)
                labels[j] = len(clusters) + 1
            end if
            clusters.append(newCluster)
        end for
        return labels, clusters
    end if
end for
end function
functiongradient(point)
return sum(exp(−(norm(point − xi)/h) ∗ ∗2) ∗ (point − xi) for xi in inputDataSet)
end function
function SAHC(point)
while True do
    gradientVector = gradient(point)
    if norm(gradientVector) < eps : then
        return point
    end if
    newPoint = point + (h/norm(gradientVector)) ∗ gradientVector
    if norm(newPoint − point) < eps : then
        return newPoint
    end if
    point = newPoint
end while
end function
Exit
```

4 Experimental Results

4.1 Data Description

In order to evaluate the various methodologies proposed, we used five data sets of different dimensions.

Algorithm 3 $DENCLUE - STHC Algorithm$

```
inputDataSet: input data set
h: density parameter
eps: stop threshold
function DENCLUE with STHC
clusters ← []
labels ←array of zeros of length equal to the size of inputDataSet
for i, dataPoint in enumerate(inputDataSet) do
    currentCluster ← STHC(dataPoint)
    if labels[i] == 0 : then
        newCluster ← [i]
        for j in range(i + 1, len(inputDataSet)) : do
            if labels[j] == 0 and distance(STHC(inputDataSet[j]), STHC(dataPoint))
    < h : then
                newCluster.append(j)
                labels[j] = len(clusters) + 1
            end if
            clusters.append(newCluster)
        end for
        return labels, clusters
    end if
end for
end function

function STHC(point)
while True do
    newPoint = None
    bestDensity = −inf
    for neighbor in neighbors(point) : do
        neighborDensity = density(neighbor)
        if neighborDensity > bestDensity : then
            newPoint = neighbor
            bestDensity = neighborDensity
        end if
    end for
    if bestDensity <= density(point) then
        return point
    end if
    point = random.choice(neighbors(point))
end while
end function

function neighbors(point)
return [point + h ∗ direction for direction in directions] + [point − h ∗ direction for
    direction in directions]
end function

function density(point)
return sum(exp(−(norm(point − xi)/h) ∗ ∗2) for xi in inputDataSet)
end function
Exit
```

- **Adult**: The "Adults" database is a collection of data produced in 1994, including full details of the demographic, social and economic dimensions of all individuals in the United States [8].
- **Lung Cancer**: This database provides information about lung cancer patients [9].
- **Iris**: Fisher's classic data set, performed in 1936. One of the first data sets used to evaluate classification methods [10].
- **Nasopharynx Morocco**: Nasopharynx cancer data set, is part of a project to identify prognostic factors for nasopharyngeal carcinoma. They were collected by the Institute Pasteur in Casablanca.
- **Nasopharynx Maghreb**: Nasopharynx cancer data set, which is part of a project to evaluate the prognostic factors of nasopharyngeal carcinoma. It combines data from patients in the Greater Maghreb region.

4.2 Validity Metrics

For a comprehensive evaluation of the results of different clustering methods, the use of specific evaluation metrics is essential. These metrics, are designed to enable accurate comparison of the performance of clustering algorithms. In other words, they provide the basis for choosing the best performing method. There are two main categories of evaluation metrics: internal and external.

4.2.1 Internal Measurements

- **DI**: It measures coherence between members of the same group, i.e. the degree to which they are similar to each other. High scores signify a better grouping quality.
- **DBI**: It measures the difference between clusters, i.e. the difference between them. Lower values indicate better clustering quality.
- **CI**: It measures the average proximity between each pair of elements in the same cluster. Its purpose is to ensure that the members of a cluster are as similar to each other as possible. A lower value of this index indicates better grouping quality.

4.2.2 External Measurements

- **CA**: This index measures the precision of grouping objects into their respective clusters, based on previously defined class labels. A higher value indicates better clustering quality.
- **NMI**: It quantifies the common statistical information between the points representing the clusters and the predefined attributes of the instances. A higher value of this index reflects better clustering quality.

- **Entropy**: It assesses the purity of each cluster by evaluating the extent to which all objects within a cluster belong to a single class. A lower value of this index indicates better clustering quality.

4.3 Comparison and Discussion

The following table shows the results of the comparison between the three versions of DENCLUE by applying the validity measures (Table 1).

- For the Adult data set, the three DENCLUE variants show distinct performances in terms of various measures. DENCLUE-SIHC stands out with the best values for DI and DBI, while DENCLUE-SAHC performs best for CI. All variants of DENCLUE achieve the same results for CA, while DENCLUE-SIHC scores highest for NMI. Entropy is highest for DENCLUE-SAHC.
- For the Lung Cancer data set, performance also varies between the three DENCLUE variants. DENCLUE-SAHC shows strong results in terms of DI and CI, while DENCLUE-SIHC achieves the best DBI value. However, when it comes

Table 1 A comparison of the three DENCLUE variants with Hill Climbing based on performance measurements

Measure	Algorithms	Adult	Lung Cancer	Iris	Nasopharynx Mor	Nasopharynx Mag
DI	DEN-SIHC	0.9508	0.4472	0.0437	0.1446	0.1125
	DEN-SAHC	0.4472	0.5	0.2611	0.0468	0.4275
	DEN-STHC	0.0	0.0001	0.05	0.125	0.0061
DBI	DEN-SIHC	0.2735	0.4369	3.9867	0.5131	0.8654
	DEN-SAHC	0.4054	0.4505	0.2633	0.1861	0.4575
	DEN-STHC	7.0150	7.4238	0.1647	0.0749	0.1527
CI	DEN-SIHC	0.2118	0.1875	0.4013	0.0411	0.1934
	DEN-SAHC	0.1281	0.1212	0.0064	0.0274	0.1686
	DEN-STHC	0.5460	0.4953	0.0015	0.0077	0.0102
CA	DEN-SIHC	0.755	0.6367	1.0	0.0384	0.9727
	DEN-SAHC	0.755	0.6367	1.0	0.0384	0.9727
	DEN-STHC	0.755	0.6367	1.0	0.0384	1.0
NMI	DEN-SIHC	0.1463	0.2083	0.5860	0.6402	0.0373
	DEN-SAHC	0.0692	0.2436	0.3665	0.6505	0.0343
	DEN-STHC	0.0674	0.1929	0.3602	0.6330	0.0635
Entropy	DEN-SIHC	1.7135	2.7279	3.8239	4.5465	4.9229
	DEN-SAHC	1.6237	3.4705	7.0637	4.4406	4.2477
	DEN-STHC	5.1710	6.5118	7.2154	4.6235	4.3092

to AC measurement, all variants show identical results. DENCLUE-SAHC again stands out with the best NMI value, while DENCLUE-SIHC offers the best entropy.

- As far as the "Iris" data set is concerned, a variation in performance is observed between the different DENCLUE versions. DENCLUE-SAHC shows a superior value for DI, while DENCLUE-STHC excels in terms of DBI and CI. For CA measurement, all three versions show equivalent results. Nevertheless, when it comes to NMI and entropy, DENCLUE-SIHC stands out with the best scores.
- In the context of the Nasopharynx data for Morocco, the DENCLUE-SIHC version delivers superior performance in terms of DI. In addition, the DENCLUE-STHC version stands out with the best values for DBI and CI. In terms of CA measurement, all three versions show similar performance. However, for NMI and entropy, the DENCLUE-SAHC version comes out on top with the most satisfactory results.
- Nevertheless, with regard to the Maghreb Nasopharynx data, it can be observed that the DENCLUE-SAHC version displays the best value for the DI index, while the DENCLUE-STHC versions stand out with the best values for the DBI and CI indices. In terms of CA and NMI, the DENCLUE-STHC version achieves the best results. For entropy, on the other hand, the DENCLUE-SAHC version performs best.

An exhaustive analysis of the performance results shows that the differences observed between the various variants of the DENCLUE algorithm can be attributed to a multitude of factors. Among these are the method used to search for local maxima, the approach chosen for estimating the probability density, the dimensions and inherent complexity of the data, and the potential presence of noise or outliers. As an illustration, DENCLUE-SIHC stands out for using a more rigorous local maxima search method than any of its homologues, which could explain its superior performance in terms of Davies-Bouldin Index (DBI) and Dunn Index (DI). Likewise, the probability density estimation method adopted by DENCLUE-SAHC is better adapted to data containing outliers, which could justify its superior performance in terms of (IC) and (NMI). Last but not least, the DENCLUE-STHC algorithm benefits from a more efficient local maxima search method, which could explain its superior results in terms of (DBI) and (CA).

An analysis of the three DENCLUE versions in terms of execution time highlights the fact that the STHC version performs the most slowly. By contrast, the two other versions perform satisfactorily in terms of execution time. Due to its stochas-

Table 2 Comparison of versions according to execution time in seconds

Data sets	DENCLUE-SIHC	DENCLUE-SAHC	DENCLUE-STHC
Adult	26.944	36.1616	468.709
Lung Cancer	64.498	94.778	1190.76
Iris	109.107	120.792	189.356
Nasopharynx Mor	4.0230	3.2844	209.25
Nasopharynx Mag	762.616	370.33	1760.108

tic approach, the stochastic version of the DENCLUE algorithm (STHC) is slower than the other versions. This is because, in stochastic methods, probability density estimation is based on the use of random data elements. This brings a higher degree of variability into the process, since the results depend on randomly selected points. Moreover, generating random points to estimate probability densities can lead to slower convergence, as the samples do not always optimally capture the underlying structure of the data (Table 2).

5 Conclusion

In this study, we examine the evolution of the DENCLUE clustering algorithm through the analysis of its different variants. Our work has carefully explored the implementation of all three versions of DENCLUE, focusing on the three types of Hill Climbing local search algorithms used in the base version and in DENCLUE 2.0. As a result of our analysis, the DENCLUE-SAHC variant emerges as the best performing of the three, demonstrating consistently strong performance across a diverse spectrum of evaluation metrics.

However, it is crucial to note that despite these improvements, the challenge presented by the slowness of execution of the DENCLUE algorithm remains. This challenge motivates us to look at new perspectives for optimization in our future research. Our aim will be to develop a new version of DENCLUE based on a combinatorial optimization approach with a population of solutions using as example approaches developed in [11–27]. This future work opens the way to potential improvements that will meet this challenge and further optimize the performance of the DENCLUE algorithm.

References

1. Fahad, A., Alshatri, N., Tari, Z., Alamri, A., Khalil, I., Zomaya, A.Y., Foufou, S., Bouras, A.: A survey of clustering algorithms for big data: taxonomy and empirical analysis. IEEE Trans. Emerg. Top. Comput. **2**(3), 267–279 (2014)
2. Hinneburg, A., Gabriel, H.-H.: Denclue 2.0: fast clustering based on kernel density estimation. In: International Symposium on Intelligent Data Analysis, pp. 70–80. Springer (2007)
3. Rehioui, H., Idrissi, A.: On the use of clustering algorithms in medical domain. Int. J. Artif. Intell. **17**, 236 (2019)
4. Rehioui, Hajar, Idrissi, Abdellah: New clustering algorithms for twitter sentiment analysis. IEEE Syst. J. **14**(1), 530–537 (2019)
5. Rehioui, Hajar, Idrissi, Abdellah, Abourezq, Manar, Zegrari, Faouzia: DENCLUE-IM: a new approach for big data clustering. Procedia Comput. Sci. **83**, 560–567 (2016)
6. Rehioui, H., Idrissi, A., Abourezq, M.: The research and selection of ideal cloud services using clustering techniques: track: big data, data mining, cloud computing and remote sensing. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies, pp. 1–6 (2016)

7. Idrissi, A., Rehioui, H., Laghrissi, A., Retal, S.: An improvement of DENCLUE algorithm for the data clustering. In: 2015 5th International Conference on Information & Communication Technology and Accessibility (ICTA), pp. 1–6. IEEE (2015)
8. Source of dataset. https://archive.ics.uci.edu/dataset/2/adult. Adult DataSet
9. Source of dataset. https://www.kaggle.com/datasets/thedevastator/cancer-patients-and-air-pollution-a-new-link. Lung Cancer DataSet
10. Source of dataset. https://archive.ics.uci.edu/dataset/53/iris. Iris DataSet
11. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
12. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Arch. **9**(2–3), 136–148 (2020)
13. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless ad hoc networks using the skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of Things and Cloud Computing (2016)
14. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
15. Elhandri, K., Idrissi, A.: Parallelization of top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
16. El Handri, K., Idrissi, A.: Comparative study of top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
17. El Handri, K., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
18. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv:1307.5910
19. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
20. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. International Conference on Big Data and Advanced Wireless Technologies (2016)
21. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
22. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
23. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
24. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
25. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular ad-hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
26. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol., pp. 5567–5584 (2023)
27. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)

Distributed DENCLUE Algorithm Based on Apache Spark

Abdellah Idrissi, Khawla Elansari, and Soukaina Elilali

Abstract Clustering is an essential technique to extract hidden patterns and similar groups from data. Consequently, clustering as an unsupervised learning method is critical in extensive data analysis, mainly due to large numbers of unlabeled data. However, efficient processing of large datasets represents a significant challenge as data volumes increase exponentially. In this context, this study aims to enhance the DENCLUE 2.0 (DENsity-based CLUstEring) algorithm, a density-based clustering algorithm, from its sequential version to a distributed version using a distributed computing approach based on Apache Spark to address this challenge and accelerate the clustering process.

1 Introduction

The rapid evolution of data globally has markedly influenced data analysis, especially in clustering methodologies. Density-based clustering methods have gained significant traction for their ability to identify clusters with varied shapes and manage noisy data effectively. The DENCLUE (Density-based Clustering) algorithm emerges as a prominent solution in density-based clustering, utilizing local density characteristics of the data space to uncover clusters with intricate shapes.

Nonetheless, the DENCLUE algorithm has drawbacks, including high computational complexity, sensitivity to parameter settings, and challenges in scalability. Its computational intensity escalates with increasing data volume and dimensionality, particularly during the density attractor identification in the Hill Climbing step. Moreover, the algorithm's performance heavily depends on the precise tuning of the density threshold (h) and noise threshold (ξ), with inappropriate parameter choices adversely affecting clustering outcomes. The algorithm also struggles with large data sets due to its demanding computational and resource requirements. This

A. Idrissi (✉) · K. Elansari · S. Elilali
Artificial Intelligence and Data Science Group, IPSS Team, Computer Science Laboratory (LRI), Computer Science Department, Faculty of Science of Rabat, Mohammed V University in Rabat, Rabat, Morocco
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_3

research addresses these limitations by leveraging distributed computing, specifically through Apache Spark, to enhance the DENCLUE algorithm's efficiency and scalability, focusing on the DENCLUE 2.0 variant.

The core challenge this study seeks to overcome is efficiently processing vast and complex datasets. Maintaining the sequential DENCLUE 2.0 algorithm's relevance becomes increasingly tricky as data grows in size and complexity. The question then becomes: How can we refine the DENCLUE 2.0 algorithm to manage massive datasets efficiently and scalably in less time?

This study's primary contribution is formulating a distributed version of the DENCLUE 2.0 algorithm that exploits Apache Spark's parallel computing capabilities to expedite the clustering of large datasets. We delve into distributed strategies using Apache Spark to boost the DENCLUE 2.0 algorithm's efficiency and scalability. We aim to address the immense data clustering challenge by proposing a distributed DENCLUE 2.0 algorithm variant that achieves precision and speed amidst the continuous data explosion.

2 Related Work

Clustering creates groups of objects (clusters) so that the objects in each cluster are distinct. In the literature, there are several types of clustering algorithms. These are generally grouped into five major categories, namely partition-based, hierarchy-based, DENsity-based, grid-based, and model-based clustering algorithms.

In this work, we pay particular attention to DENsity-based clustering algorithms. This family of methods has proven its effectiveness in clustering thanks to its ability to find clusters of arbitrary shapes and detect noisy objects. In this context, we focus on the DENCLUE 2.0 algorithm.

DENsity-based clustering is an approach that differs from traditional data clustering methods. Instead of dividing data into predefined clusters, this method identifies clusters based on the density of data points in the data space, enabling clusters of arbitrary shapes and sizes to be determined and even handling data containing noise, as shown in Fig. 1.

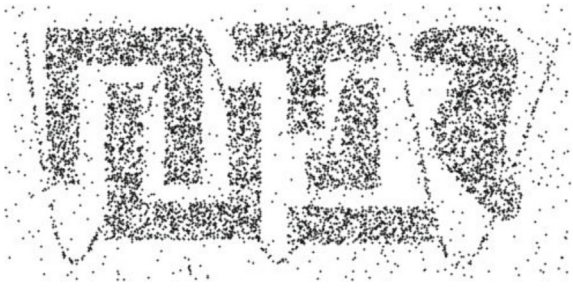


Fig. 1 Arbitrary shape data [5]

The critical steps of the DENCLUE algorithm involve utilizing density functions to locate high-density regions that define clusters. Here's an overview of the main steps involved in the DENCLUE algorithm:

- Density Estimation: The first step is to estimate the density at each point in the dataset. This is usually done using a kernel density estimation method, where the density at a point is calculated as the sum of influences from nearby points. The influence of a point decreases with distance, and a kernel function and a bandwidth parameter control this effect.
- Identify Local Density Attractors: A local density attractor is a point in the data space where the density is locally maximal. These points are identified by applying a hill-climbing algorithm that starts from a data point and moves toward increasing density until a local maximum is reached. Each local density attractor represents a potential cluster center.
- Hill Climbing: For each point in the dataset, the hill-climbing process is applied to find the nearest local density attractor. This step effectively assigns each point to a local maximum, thereby grouping points in the same dense region.
- Form Clusters: After all points have been assigned to local density attractors, clusters are formed by grouping points with the same local density attractor. Points not belonging to any dense region (i.e., their density is below a specified threshold) are considered noise.
- Handle Noise: DENCLUE defines a threshold for the minimum density required for a point to be considered part of a cluster. Points with a density below this threshold are classified as noise and are not assigned to any cluster.
- Determine Cluster Borders: Optionally, the algorithm can also determine the borders of clusters by analyzing the gradient of the density function. This involves identifying areas where the density gradient points away from local density attractors, indicating the edges of clusters.

2.1 Clustering Steps of DENCLUE 2.0

The general flow of the DENCLUE 2.0 algorithm is summarized in Table 1 [3].

The DENCLUE 2.0 algorithm performs clustering by estimating the density around each point in a data set. It uses the optimization process, Hill Climbing,

Table 1 Description of the general flow of the DENCLUE algorithm

Step 1	Calculating the density function of each point
Step 2	Identifying the points that are local maxima, usually called density attractor
Step 3	Defining clusters, which consist of points associated with a particular density attractor
Step 4	Discarding clusters whose density attractor has a density less than ξ
Step 5	Combining clusters that are connected by a path of points that all have density of ξ or higher

to find attractive positions and potential clusters. Valid clusters are then selected according to their density, and final labels are assigned to samples based on their membership of the identified clusters. The DENCLUE 2.0 algorithmic process is shown in Algorithm 1.

Algorithm 1: Sequential DENCLUE 2.0 Algorithm

Input : Data points $D = \{ p_1, p_2, \ldots, p_n , \}$ Bandwidth h, threshold ξ
Output: Data points labeled, Cluster ID = 0
for *each point p in D* **do**
Calculate Density Attractors using the Hill Climbing process. Create Clusters Graph
Create an empty graph G
for *each data point p in D* **do**
Recuperate its density attractors, radius, and density. Add node to the graph G
for *each data point* p_1 *in D* **do**
for *each data point* $p_2 = /p_1$ *in D* **do**
Calculate the distance between their density attractor spaces. Calculate the the sum of their radius **if the** *distance of attractors < sum of the radius,* **then**
Add edge between p_1 and p_2 in G

Identify Connected Components in Graph G Each connected component corresponds to a potential cluster
for *each sub-cluster in graph G* **do**
Find the data point with the highest density attractor (max_density)
if *max_density >= threshold* ξ **then**
Assign a label to this point and all attracted points
else
Assign cluster ID = -1

Output: cluster labels assigned

However, the DENCLUE 2.0 algorithm has limitations in terms of parameter sensitivity as the selection of these parameters significantly influences the quality of the clustering results in terms of computational complexity, as the DENCLUE 2.0 algorithm is computationally time-consuming during clustering due to the calculations for each data point to find density attractor in the Hill Climbing step [4] and also in terms of scalability, as the DENCLUE 2.0 algorithm is sensitive to large datasets due to its complexity and computational resource requirements.

3 Contribution

3.1 Distributed DENCLUE 2.0 Algorithm Based on Spark

Our approach is to distribute the DENCLUE 2.0 algorithm to speed up the execution process compared with its sequential version based on the Hill Climbing algorithm. This crucial step in the DENCLUE 2.0 algorithm involves calculating each point's density attractor. However, performing these calculations for each point becomes impractical when working with large databases.

Apache Spark is one of the most popular and powerful frameworks in this category for distributing computational tasks across a cluster of connected machines to harness the power of parallel processing to analyze massive data. Spark is based on a distributed architecture that leverages the computing power of multiple nodes within a cluster.

Exploiting the advantages of the Apache Spark Distributed Computing Framework in distributing the computational load across multiple processing nodes, we present in Fig. 2 Our proposed solution for our approach to the distributed DENCLUE 2.0 algorithm. The general idea is to divide the data into smaller subsets, called partitions, which are then processed in parallel on different distributed system nodes. Each node runs an instance of DENCLUE 2.0 on its data partition, calculating local densities and merging clusters locally. Then, the partial results of the different DENCLUE 2.0 instances are combined to obtain the global clusters.

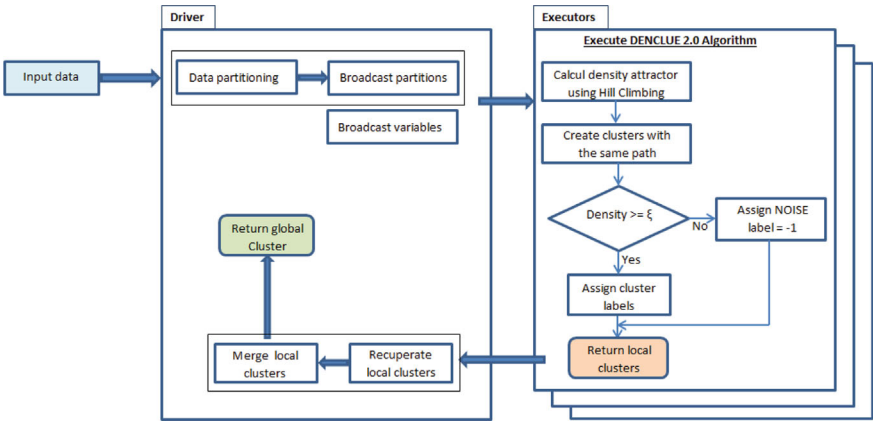


Fig. 2 Diagram of the DENCLUE 2.0 distributed algorithm

3.2 Implementation

In developing our distributed version of the DENCLUE 2.0 algorithm, we selected two datasets, namely Iris and Ecoli, for testing. Initial data preparation steps included removing any missing values, normalizing the datasets, and adjusting specific attributes to align with the requirements of the DENCLUE 2.0 framework. To execute the DENCLUE 2.0 algorithm in a distributed manner, we set up and tailored the Spark environment on a local system to meet our needs. To determine the optimal settings for the DENCLUE 2.0 algorithm, specifically the bandwidth (h) and the noise threshold (ξ), we employed the Optuna framework to fine-tune these parameters for each dataset. The distributed execution of DENCLUE 2.0 is structured into three main phases: splitting the data into partitions, conducting clustering locally within each partition, and finally merging these local clusters into a global overview. These phases are elaborated upon in subsequent sections.

3.2.1 Data Partitioning

In a distributed Spark environment, workload distribution, and processing efficiency are optimized through various data partitioning strategies. These strategies include data region partitioning, which is primarily divided into three popular methods: Even Split Partitioning (ESP), Reduced Boundary Points (RBP), and Cost-based Partitioning (CBP). For our project, we have implemented the ESP method of data region partitioning, as demonstrated in Fig. 3, on our selected datasets, Iris and Ecoli.

The entire dataset is viewed within this framework as a single global partition, referred to as Partition P. This global partition is then segmented into multiple distinct partitions based on defined criteria. To maintain a connection between these partitions, boundary regions are established. Consequently, two adjacent partitions, Partition1 and Partition2, are expanded to include boundary regions. Data points within

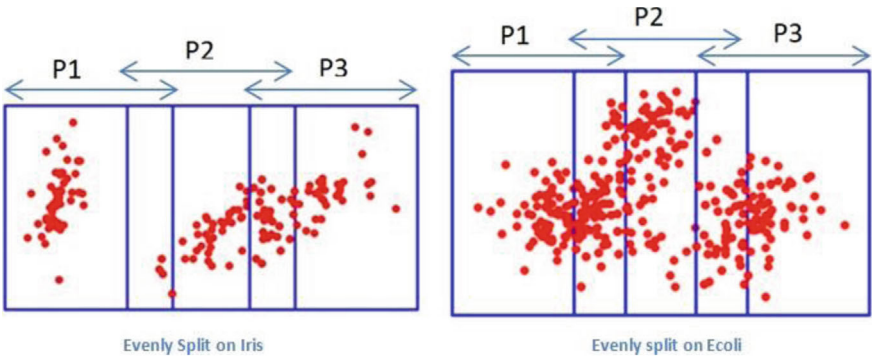


Fig. 3 Example of the application of evenly split partitioning to different datasets with three partitions

these boundary areas are duplicated and exist in Partition1 and Partition2. This duplication approach facilitates the identification of neighboring points while merging local clusters.

The data partitioning process involves three key steps: first, we divide the data space of the two datasets, iris and ecoli, into three partitions, then we extend each partition by a margin, allowing adjacent partitions to share boundary points, then each point is assigned to the corresponding partition according to its coordinates.

3.2.2 Local Clustering

The local clustering part focuses on running the DENCLUE 2.0 algorithm locally on RDDs partitioned in advance. This step takes the sequential DENCLUE 2.0 algorithm mentioned in Algorithm 2 and adapts it to run in parallel. This step aims to obtain local clustering results. To achieve this, the DENCLUE 2.0 clustering algorithm is run independently on each partition, taking full advantage of Spark's parallel computing capabilities, thus distributing the workload evenly and considerably improving the algorithm's overall performance.

In this step, points belonging to the same partition are processed together. It is worth noting that the broadcast() method distributes the critical variables of DENCLUE 2.0 to all partitions, namely the smoothing parameter h and the threshold parameter ξ. This means that a copy of each variable is used on each partition, and there is no need to transfer variables for each task. This reduces transmission time and improves the overall clustering performance. Algorithm 2 outlines the pseudo-code used to perform local clustering. This pseudo-code details the specific steps of the DENCLUE 2.0 algorithm adapted to our parallel approach based on Spark.

In summary, this local clustering section ensures that the DENCLUE 2.0 algorithm operates efficiently and in parallel across the entire partitioned dataset, thereby optimizing the overall performance of the distributed clustering algorithm DENCLUE 2.0.

Algorithm 2: Distributed DENCLUE 2.0 Algorithm

Input: D: dataset
Input: h: bandwidth
Input: ξ: Threshold
Input: eps: convergence parameter
Input: n: number of partitions
for each partition n **do**
    Broadcast h, ξ, and eps to partition n;
    denclue(D) // Exécution DENCLUE sequential sur chaque partition;
Output : list (p_ID, is_core), labels // clusters locaux de chaque partition;

The DENCLUE 2.0 algorithm begins by assigning an initial density attractor to each data point, which serves as the starting position for the kernel optimization process to find the final density attractor. This optimization aims to discover a new

position for the density attractor around each data point where the local density is maximized.

As the algorithm progresses and the optimization process iterates for each data point, density attractors evolve and move towards positions where local density is maximized.

The algorithm updates the density attractors until convergence is achieved for each data point, meaning that the density attractors no longer move significantly. The final density attractors are obtained when the kernel optimization process converges for each data point.

Subsequently, the algorithm constructs clusters by connecting overlapping density attractors, i.e., those whose radius overlap significantly. Density attractors sufficiently close to each other are considered to form the same cluster, and data points whose density attractors do not overlap significantly remain isolated in their clusters. Therefore, each partition returns the local clusters.

3.2.3 Global Merge

In the final step of the DENCLUE 2.0 clustering phase, we merge the local clusters to form global clusters. Our fundamental approach relies on combining the results from each partition using the previously mentioned intersection points for double verification.

The phase of merging local labels into global labels for the entire dataset is a crucial step in the distributed clustering process. The main goal of this function is to merge the local labels obtained from the local clustering of each partition in the previous step into global labels while maintaining label consistency across the dataset.

The merging process outlined in Algorithm 3 proceeds as detailed in Fig. 4.

First, we obtain the RDD result from the previous local clustering step. We extract each data point's identifier p_id, its point type (whether it's a Core point, i.e., is_core equals 1, or not, i.e., is_core equals 0), and its local label.

Next, we check if the point is new during the merging process. If it is new, it is added to the resulting RDD. However, if the data point already exists and is a Core point, in that case, all points attracted to this point will be assigned to the corresponding label and added to the final RDD result, marking the point as globally labeled. If the data point has yet to be labeled globally, we update its global label and mark it as labeled. This way, local results are gradually incorporated into the final result and are ready to be used in subsequent output and collection steps.

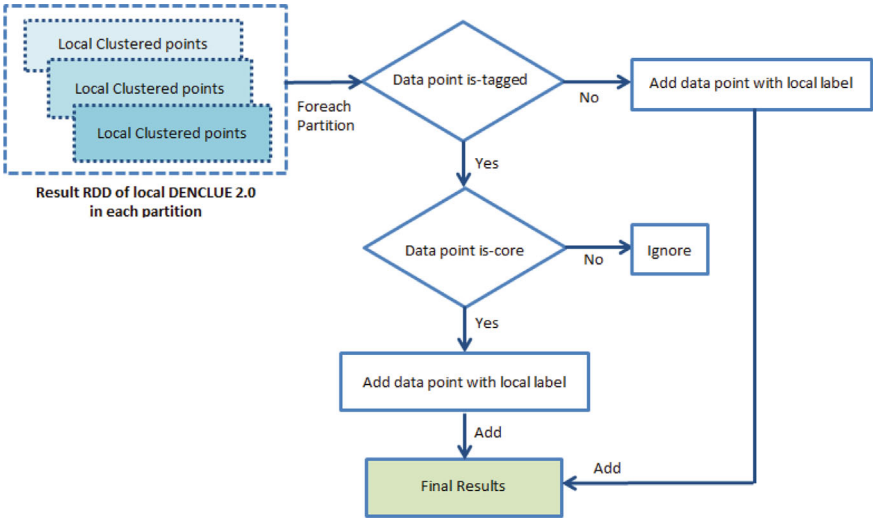


Fig. 4 Flowchart for merging local results

Algorithm 3: Merge local results

Input: list (p_ID, is_core), labels
Output: global clusters
for each partition *Pi* in partitions *P* **do**
  for each point *p* in partition *Pi* **do**
    if point *p* is tagged and is core point **then**
      Update the local label of all its neighbors with the global label;
      Put into the result RDD;
    else
      Update the local label of *p_ID* by the global label and is tagged;
Output result: global clusters

4 Results and Discussion

4.1 Dataset Description

To evaluate our approach, we utilized two real datasets of different sizes, each with distinctive characteristics. Table 2 presents the features of these datasets.

Table 2 Dataset information

Datasets	Observations	Attributes	Classes
Iris	150	4	3
Ecoli	336	8	8

4.2 *Comparison of DENCLUE Algorithm Versus Distributed DENCLUE Algorithm*

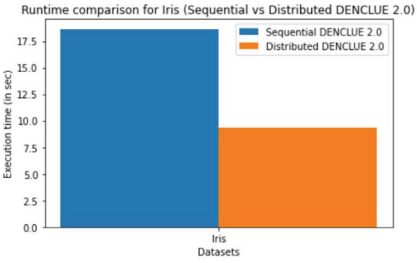
4.2.1 Execution Times

The execution time is a measure that indicates the total duration required to complete the execution of a computer program. It is often used to assess the performance and efficiency of a process or application. The execution time (in milliseconds) of the sequential and the parallel DENCLUE 2.0 algorithm is mentioned in Table 3 (see Fig. 5).

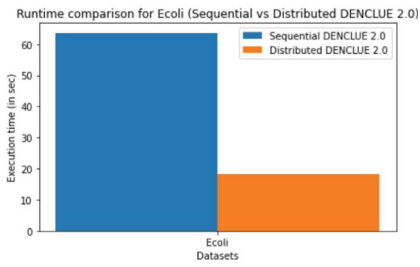
The results demonstrate improved performance using the distributed DENCLUE 2.0 algorithm compared to its sequential version. We observe that, for the Iris dataset, the execution time decreased from 18.64 seconds with the sequential DENCLUE 2.0 algorithm to only 9.35 s with the distributed DENCLUE 2.0 algorithm. Similarly, for the Ecoli dataset, the execution time decreased from 63.65 s with the sequential DENCLUE 2.0 algorithm to only 18 s with the distributed DENCLUE 2.0 algorithm. Notably, the distributed version of the DENCLUE 2.0 algorithm proves to be time-saving compared to the sequential DENCLUE 2.0 algorithm. This result can be explained by using a specific number of executors and partitions in the distributed version of the algorithm, ensuring that the data volume for each partition is smaller. This reduction in data volume per partition leads to a corresponding decrease in time costs for each partition, resulting in an overall shorter execution time. In contrast,

Table 3 DENCLUE 2.0 sequential versus DENCLUE 2.0 distributed runtimes

Datasets	Sequential DENCLUE 2.0 (in s)	Distributed DENCLUE 2.0 (in s)
Iris	18,64	9,35
Ecoli	63,65	18,09



(a) DENCLUE 2.0 sequential Vs. DENCLUE 2.0 distributed for the Iris dataset



(b) DENCLUE 2.0 sequential Vs. DENCLUE 2.0 distributed for the Ecoli dataset

Fig. 5 Comparison between sequential DENCLUE 2.0 and distributed DENCLUE 2.0 for different datasets

the sequential version of the DENCLUE 2.0 algorithm operates independently of the number of executors and runs sequentially across the entire dataset.

- **Speedup**

Speedup is a performance metric that assesses the efficiency of a parallel system or program when executed on multiple processors or computing cores compared to its sequential execution on a single processor. It represents the speed gain achieved by running on multiple processors. Speedup is typically calculated using the following formula:

$$\text{Speed up} = \frac{T_{\text{Seq}}}{T_{\text{P}}}$$

where T_{Seq} is the execution time required for the sequential approach, and T_{P} is the execution time needed to run the same program on a parallel system. Speedup measures how much faster parallel execution is compared to sequential execution. A speedup greater than 1 indicates performance improvement through parallelization. The higher the speedup, the more significant the performance improvement (see Fig. 6 and Table 4).

We observe that the parallel approach allows processing at approximately two times the speed compared to the sequential approach for the same dataset. Similarly,

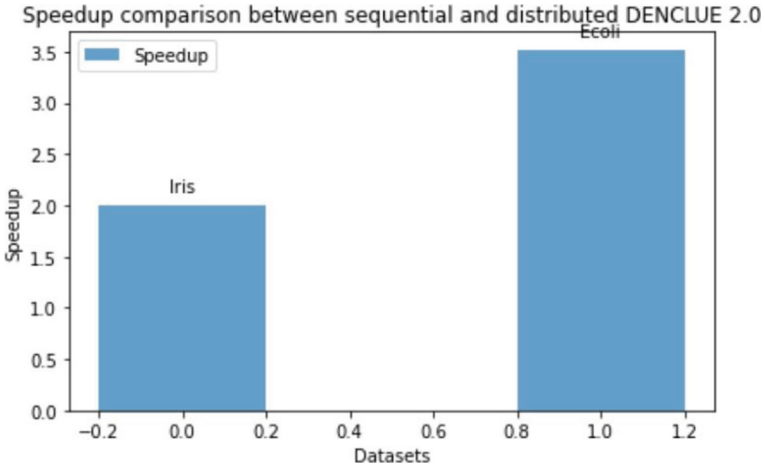


Fig. 6 Speed up between sequential DENCLUE 2.0 and distributed DENCLUE 2.0

Table 4 Table of Speedup values

Datasets	Speedup
Iris	1,99
Ecoli	3,52

for the Ecoli dataset, the parallel approach enables processing at approximately 3.5 times the speed compared to using the sequential approach for the same dataset.

5 Conclusion and Prospects

This paper has provided a detailed overview of our research focused on the distributed version of the DENCLUE 2.0 algorithm through a distributed approach based on Apache Spark. We have covered various stages, from exploring basic concepts to practical implementation and conducting experiments to evaluate our approach. Our contribution has been extensively presented. We discussed data partitioning, parallel execution of local clustering, and merging local results to obtain global clusters and described the specific methods and applied approaches to understand our work thoroughly.

In conclusion, our research has led to the development of a distributed version of the DENCLUE 2.0 algorithm, leveraging the capabilities of Apache Spark to accelerate the clustering of large datasets. The results of our experiments suggest that our approach provides a significant performance improvement compared to the sequential DENCLUE 2.0 algorithm. However, challenges remain, and future work could focus on optimizing our method and adapting it to more complex scenarios.

In the context of future directions and developments, the following areas can be considered:

- Improving parallelization efficiency: Our project was primarily executed on a local machine, and to harness the power of distributed computing, we plan to explore execution on much more powerful real machines to process even more complex datasets.
- Advanced partitioning and merging strategies: The crucial importance of choosing the partitioning strategy in the clustering process prompts us to apply advanced partitioning strategies to ensure a balanced distribution of workloads among workers, thereby optimizing the algorithm's overall performance. Additionally, we plan to enhance the merging phase of local results by exploring more reliable and efficient techniques.
- Dynamic Clustering: In the long term, developing a dynamic clustering algorithm. This algorithm should be capable of adapting to incoming data dynamically and in real-time, providing a more flexible and responsive solution to the changing needs of the data.

In summary, this study makes a significant contribution. It paves the way for further in-depth research that can be undertaken to assess the performance of distributed DENCLUE 2.0 in various application scenarios. We intend to apply this manner of distribution to our different methods conceived for several applications like those in [6–22]. However, additional research and development efforts are needed to achieve its potential and fully address the identified limitations.

References

1. Hinneburg, A., Gabriel, H.-H.: DENCLUE 2.0: fast clustering based on kernel density estimation. In: Berthold, M.R., Shawe-Taylor, J., Lavrac, N. (eds.) Advances in Intelligent Data Analysis VII. Lecture Notes in Computer Science, vol. 4723, pp. 70–80. Springer Berlin Heidelberg (2007)
2. Luo, Y., Zhang, K., Chai, Y., Xiong, Y.: Multi-parameter-setting based on data original distribution for DENCLUE optimization. IEEE Access **6**, 16704–16711 (2018)
3. Musdholifah, A., Hashim, S.Z.M.: Cluster Analysis on High-Dimensional Data: A Comparison of Density-Based Clustering Algorithms (2013)
4. Rehioui, H., Idrissi, A., Abourezq, M., Zegrari, F.: DENCLUE-IM: a new approach for big data clustering. Procedia Comput. Sci. **83**, 560–567 (2016)
5. Ertoz, L., Steinbach, M., Kumar, V.: A new shared nearest neighbor clustering algorithm and its applications. In: Workshop on Clustering High Dimensional Data and Its Applications at 2nd SIAM International Conference on Data Mining, vol. 8 (2002)
6. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
7. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv: 1307.5910
8. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
9. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
10. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
11. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
12. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
13. Elhandri, K., Idrissi, A.: Comparative study of Top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
14. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
15. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
16. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
17. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
18. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
19. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
20. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)

21. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
22. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)

Distributed Evostream Algorithm Based on Apache Spark

Abdellah Idrissi, Khawla Elansari, and Mahmoud Lham

Abstract Clustering is pivotal in data mining, aiming to uncover hidden structures within datasets. It is extensively applied in fields like marketing and healthcare for grouping similar entities. With the advent of large-scale, rapid data sources like sensors, clustering these vast and dynamic data streams has become increasingly significant. The evoStream algorithm stands out for its capability in data stream clustering. This study delves into crafting a distributed variant of the evoStream algorithm in Python, explicitly focusing on distributing its offline phase—an evolutionary optimization technique utilized across various domains. To facilitate this distribution, the research integrates the principles of the "Master–Slave Model," employing a master–slave architecture, and the "Island Model," which segments the process into multiple independent units or "islands." Each Island executes a version of the genetic algorithm tailored with unique parameters and populations. The primary aim is to evaluate the performance of the distributed evoStream algorithm, as designed using the Island Model, against the traditional, single-machine sequential evoStream algorithm. This evaluation is based on metrics such as execution time and algorithm speedup. The dissertation meticulously examines the outcomes of this comparison, shedding light on the strengths and weaknesses of both methodologies. It offers valuable insights for scholars and practitioners working on optimization with evoStream.

1 Introduction

In today's Big Data landscape, the surge in real-time data generation across multiple sectors, including social media, the Internet of Things (IoT), surveillance, healthcare, and others, has underscored the necessity for immediate data analysis. This real-time analysis is crucial for quick decision-making, trend spotting, anomaly detection, and process optimization.

A. Idrissi (✉) · K. Elansari · M. Lham
Artificial Intelligence and Data Science Group, IPSS Team, Computer Science Laboratory (LRI), Computer Science Department, Faculty of Sciences of Rabat, Mohammed V University, Rabat, Morocco
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_4

To address this need, real-time data clustering algorithms, which group similar data points into clusters to uncover patterns within data streams, have become invaluable. Among these, the evoStream algorithm, introduced by Carnein and Trautmann [1], is noteworthy for its evolutionary approach to data stream clustering. Drawing on evolutionary algorithms, notably the genetic algorithm, evoStream is designed to adapt dynamically to ongoing data changes, ensuring the continuous and accurate updating of clusters to reflect the evolving nature of streaming data.

Nonetheless, the challenge of scaling up to handle burgeoning data volumes without compromising real-time performance is significant. The sequential processing model of evoStream is increasingly unable to meet the demands of large-scale data analysis, leading to delays that can hinder real-time analytics and scalability.

To mitigate these challenges, developing a distributed version of the evoStream algorithm in Python is proposed as a strategic move. This approach aims to enhance the algorithm's capacity for handling real-time data stream analysis challenges, such as reducing latency and increasing processing speed. By distributing the workload, the algorithm can offer robust solutions for anomaly detection, process optimization, and intelligent decision-making across diverse sectors, including industry, finance, healthcare, and more. Additionally, implementing evoStream in Python will enhance its accessibility and ease of integration into various data streaming architectures, broadening its applicability and relevance in the dynamic domain of real-time data analysis.

2 Related Work

2.1 Evolutionary Algorithms

Drawing inspiration from Darwin's theory of evolution, evolutionary algorithms are search and optimization strategies that mimic the mechanisms of biological evolution to address complex problems. These algorithms operate on a fundamental process, as depicted in Fig. 1, starting with generating a diverse population of individuals. Each individual's performance is evaluated using a fitness function, identifying those most suited to the problem. The fittest individuals are then chosen for reproduction, undergoing crossover and mutation to produce offspring. These offspring form the next generation, potentially possessing better solutions due to the genetic variations introduced. This selection, reproduction, and mutation cycle continues, iteratively refining the population until a solution meeting the desired criteria emerges [2, 3].

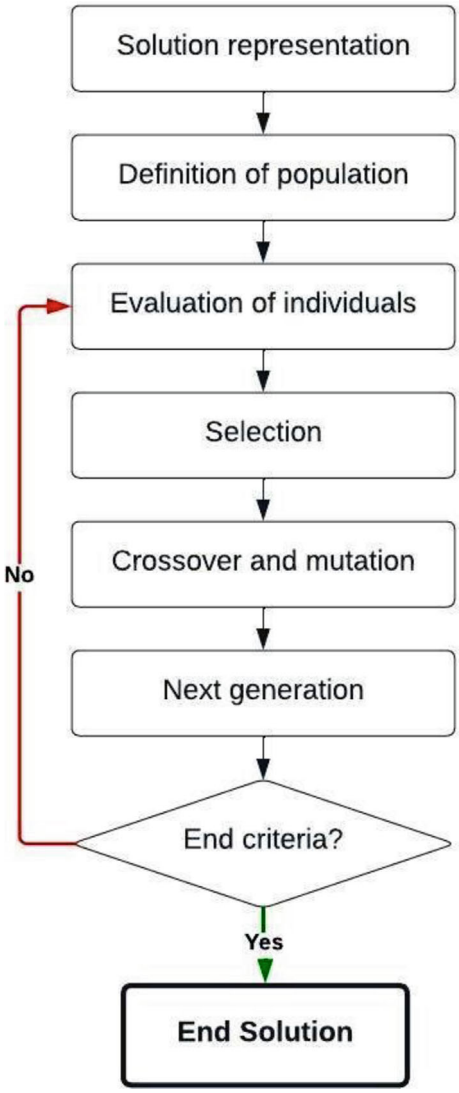


Fig. 1 General diagram of an evolutionary algorithm

2.2 Data Stream Clustering

Data stream clustering is distinct from traditional clustering methods [4] because it deals with data streams that are in a state of constant flux. These data streams can emerge swiftly and in potentially infinite volumes. As such, clustering algorithms designed for data streams must efficiently manage memory and resources, adapt to dynamic changes within the data, and offer real-time results with negligible delay.

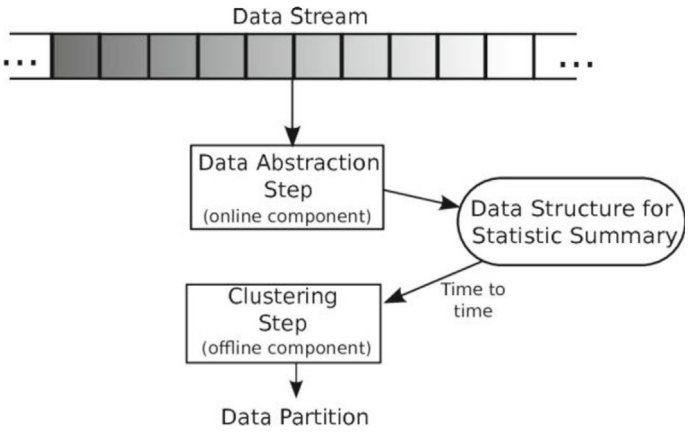


Fig. 2 Framework for clustering based on object data flow [5, 6]

The process of data stream clustering can be segmented into two principal phases, as depicted in Fig. 2 from [5]:

- **Online Phase** (or "Online Component"): In this initial phase, the focus is condensing the streaming data into compact data structures. This condensation helps address the spatial and memory limitations of streaming applications.
- **Offline Phase** (or "Offline Component"): This subsequent phase involves applying conventional clustering techniques on the condensed data to discover patterns or clusters of similar items. The clusters identified are then dynamically updated in response to each new data input.

The primary function of data stream clustering algorithms is to ensure the ongoing processing of data while adjusting to any shifts in the data's distribution, thereby enabling the delivery of clustering outcomes in real time or with minimal latency.

2.3 evoStream Algorithm

The evoStream algorithm, introduced by Carnein and Trautmann [1], represents an innovative method for clustering data streams, offering substantial improvements over traditional clustering techniques. This method draws upon genetic algorithms, a subset of evolutionary algorithms inspired by the process of natural selection, to evolve populations of potential solutions. In this context, a population of individuals evolves to produce new generations through reproduction. Selection processes mimic natural selection, ensuring only the most well-adapted individuals survive. Offspring are generated through crossover, combining traits from two parents, and mutations introduce random variations in the next generation's population traits.

This algorithm operates in two main phases, as depicted in the illustrations. Initially, data is organized into a grid structure before being further classified into three major clusters.

– During the online phase, the algorithm deals with incoming data points, establishing micro-clusters that preliminarily categorize the stream's data. This phase involves initializing new micro-clusters with specific attributes, aggregating these micro-clusters to summarize the data stream effectively, and periodically cleaning up to remove underpopulated clusters. Once a certain number of observations have been processed, macro-clusters are formed, each composed of multiple micro-clusters.
– The offline phase focuses on refining these macro-clusters using genetic algorithms akin to the GA-clustering algorithm. This includes evaluating the fitness of each solution by assessing the sum of squared distances between micro-cluster centers and their nearest macro-cluster center, selecting pairs of solutions based on fitness for recombination, and implementing crossover and mutation operations to generate and mutate offspring. These new individuals may replace the least fit individuals in the population, enhancing the quality of clusters over time.

The limits of the evoStream algorithm:

- The number of final clusters is given as an input parameter, which may lead to incorrect results in non-uniform flows.
- The algorithm uses a genetic algorithm in the offline phase, which can be expensive regarding computational resources.
- If data generation in the stream is slow, the offline phase may become idle and wait for the following observation, wasting system resources.

To deal with these limits, we will design a distributed version of the evoStream algorithm based on the Apache Spark Framework.

3 Contribution

Upon evaluating the sequential implementation of the evoStream algorithm, we observed a noticeable delay in evolutionary processing as data volume increased, along with the computational demands of the genetic algorithm used in the second phase of the algorithm. To mitigate this delay, we implemented the Apache Spark Framework algorithm, enabling the distribution of this phase across multiple nodes to enhance processing speed.

Our exploration of various literature revealed different distribution strategies for genetic algorithms, including the master–slave, Island, and Cellular models. For our purposes, we decided to adopt the first two models.

3.1 Apache Spark Framework

Apache Spark is constructed around an architecture with several components designed to ease development and ensure efficient, high-speed processing. Created initially to overcome the limitations of the MapReduce Hadoop Framework, Spark has evolved into a comprehensive platform that extends beyond simple batch processing.

In Spark, operations revolve around creating and manipulating distributed collections of data, known as Resilient Distributed Datasets (RDDs). Spark manages the distribution of RDD data across the cluster and parallelizes operations for efficient processing. RDDs support two kinds of operations: transformations, such as map(), flatMap(), and filter(), which generate new RDDs from existing ones, and actions like collect() and count(), which compute results and send them back to the master node.

A key feature of Spark is its lazy evaluation approach. This means that the operation is not executed immediately when a transformation is applied to an RDD (e.g., map()). Instead, it is deferred until an action is invoked, optimizing the execution process.

3.2 Proposed Approach

To enhance the processing speed of the offline phase, we propose a distributed implementation of the evoStream algorithm utilizing the Apache Spark Framework.

This approach, as depicted, involves distributing the macro-clusters formed during the initial phase across various partitions as an RDD. Within each partition, operations of the genetic algorithm and fitness evaluations are conducted independently from those in other partitions. In line with the practices from our sequential implementation of evoStream, we employed roulette wheel selection, uniform crossover, exchange mutation, and a selection strategy favoring the survival of the fittest parents for generating new offspring for subsequent generations.

To clarify the distributed version of the evoStream algorithm, we've created a simplified process diagram depicted in Fig. 3. This diagram aims to streamline the steps in the distributed evoStream algorithm visually.

3.3 Pseudo Code

In this approach, we aimed to distribute the second phase of the sequential evoStream. The pseudo-code is developed in the Algorithm 1. The macro-clusters are distributed between (m) partitions at line (7). At lines (8), (9), and (10), the micro-clusters, the

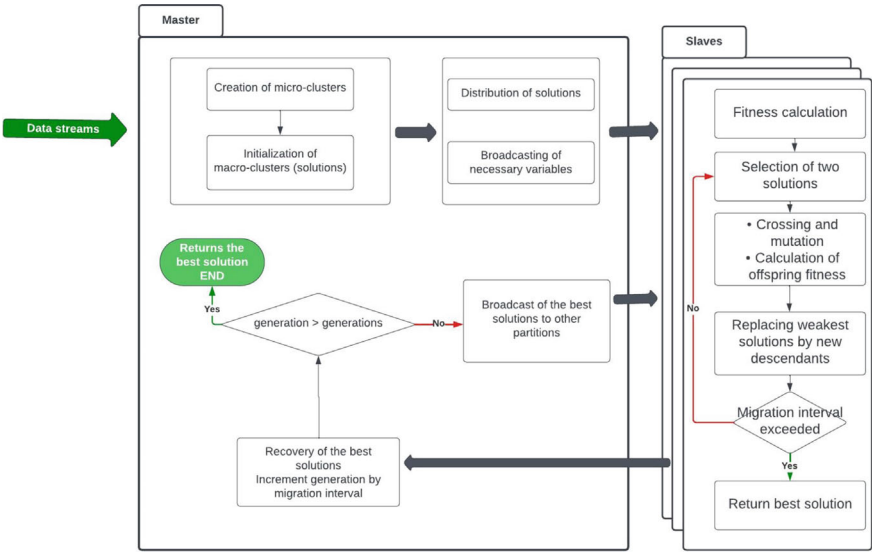


Fig. 3 DevoStream—distributed approach process diagram

migration interval (Mi), and the number of individuals to migrate (Ms) are broadcast to all partitions.

The solutions are improved using the evolution() function in line (15). It is worth mentioning here that we used operations that calculate and sort the fitness within each partition using Apache Spark's map- PartitionsWithIndex() transformation function, helping to reduce communication overhead and obtain efficient performance.

In line (17), the algorithm broadcasts the best-evolved solutions to the other partitions, and the weak solutions of the partitions are replaced by the new solutions broadcast in line (20). The migration interval (Mi) defines the number of generations after which distributed evoStream broadcasts the best individual from each partition to the other partitions. This helps ensure diversity in each subpopulation while seeking the best solutions. The number of individuals released and the migration interval affect the algorithm's running time.

Algorithm 1 EvoStream's distributed approach algorithm

```
Pi: Sub-Population at partition i
m: Number of Partitions
Mi: Migration Interval
Ms: Migration Size
G: Generations
Solutions Of size P
Distribute P between m partitions
Broadcast Micro-clusters                ▷ Broadcasting micro-clusters between m partitions
Broadcast Migration Interval Mi         ▷ Broadcasting Mi between m partitions
Broadcast Migration Size Ms             ▷ Broadcasting Ms between m partitions
g ← 0
while g < G do
    Begin at each partition i
        for k ← 1 to Mi do
            evolution()                 ▷ execution of the genetic algorithm operations in each partition
        end for
        Broadcast Ms solutions          ▷ Gathering the best solutions from each partition and
                                          broadcasting them to the other partitions
        End at each partition i
    Pi ← (Pi - (Weakest (Ms)Solutions)) ∪ BroadcastSolutions
    g ← g + Mi
end while
```

Algorithm 2 evolution() Function

```
function evolution(·)
    p1, p2 → Select two solutions proportionally to their fitness from C
    o1, o2 → Create offsprings of p1, p2 using binary crossover
    for each gi in o1, o2 do                    d For each child-gene
        if random(0,1) < Pm then                d Mutate with probability Pm
            if gi = 0 then
                gi → 2δ
            else
                gi → 2δ · gi
    Add o1, o2 to C and discard the two least fittest solutions
```

4 Results and Discussion

4.1 Measurements

We will use two measurements to evaluate the performance of the sequential version of the evoStream algorithm and its proposed distributed version: processing acceleration (Speedup) and execution time.

4.2 *Comparison of evoStream Algorithm Versus Distributed evoStream Algorithm*

The following table presents the execution time of the sequential and distributed versions of the evoStream algorithm about the number of generations and the migration interval (see Table 1).

– **Execution times**

The execution time of the sequential evoStream algorithm increases by increasing the number of generations, as shown in Fig. 4, and this evolution of latency is due to memory consumption.

Unlike the sequential approach, it is observed that the proposed distributed version offers better results with low latency, as illustrated in Fig. 5. Remarkably, when increasing the migration interval leads to a reduction in latency. This could be explained by the fact that expanding the migration interval implies decreasing the number of generations at the master level.

Table 1 Performance comparison between sequential and distributed evostream algorithm

Number of generations	Sequential evoStream execution time (s)	Distributed evoStream		Speedup
		Migration interval	Execution time (s)	
100	24,47	1	715,1	0,03
		10	58,91	0,42
		20	28,39	0,86
		25	24,43	1,00
1000	128,59	1	1500	0,09
		100	53,6	2,40
		200	33	3,90
		250	29,31	4,39
4000	598,72	100	196,5	3,05
		250	85,03	7,04
		500	54,87	10,91
		1000	39,16	15,29
10,000	1436,7	500	135,3	10,62
		1000	79,3	18,12
		2000	58,75	24,45
		4000	44,63	32,19

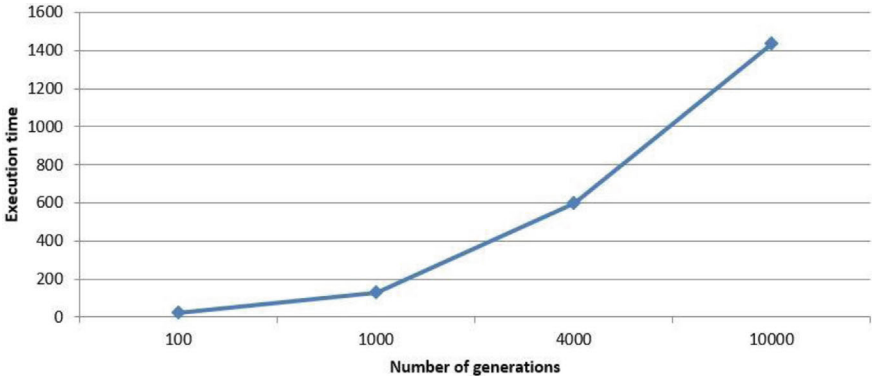


Fig. 4 Execution time according to the number of generations for the sequential evoStream algorithm

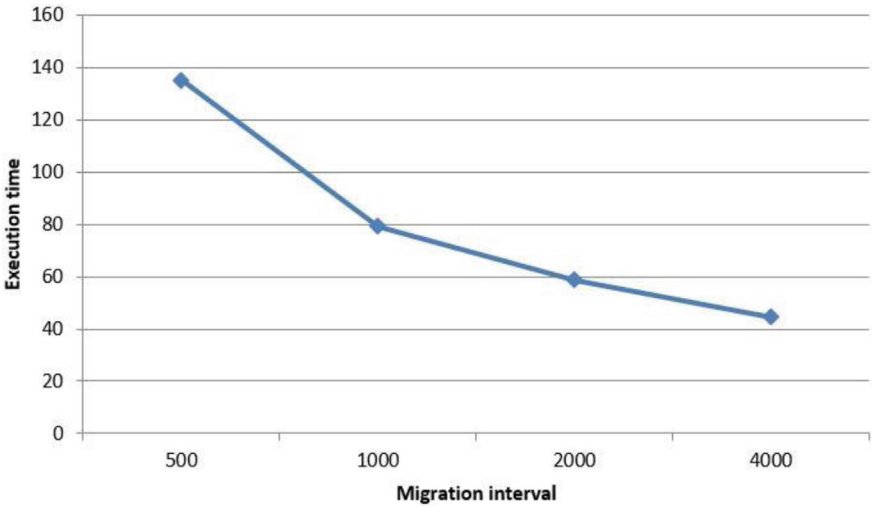


Fig. 5 Execution time according to the migration interval for the distributed evoStream algorithm (number of generations = 10,000)

– **Speedup**

From the data presented in the table above, we notice a significant increase in the speedup of the evoStream algorithm in its distributed version compared to its sequential version. This is demonstrated by the visual results shown in Fig. 6, where the speedup is highlighted.

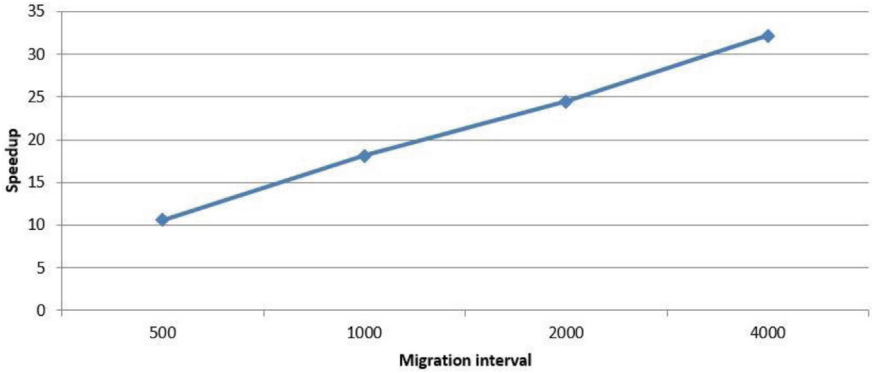


Fig. 6 Speedup according to the migration interval for the distributed evoStream algorithm (for generation number = 10,000)

From this figure, we conclude that the speedup in the distributed approach is strongly correlated with the migration interval, which makes this parameter essential in the distributed version of the evoStream algorithm.

5 Conclusion and Prospects

This research paper was an enriching and instructive experience. The main objective of this study was to develop a distributed version of the evoStream algorithm by exploiting Apache Spark under Python, focusing on processing acceleration and execution time.

This research study allowed us to discover the importance of evolutionary algorithms and distributed computing, two areas that are growing exponentially in the world of computing.

We explored in detail the concepts, tools, and libraries associated with these areas, and we understood why distributed computing has become essential for efficiently processing large amounts of data.

Our results are consistent with the objectives set; the distributed approach of the evoStream algorithm proved to be more efficient than its sequential version. We saw a significant speedup during processing and a noticeable reduction in execution time using Apache Spark.

This result demonstrates the effectiveness of distributed computing in improving the performance of evolutionary algorithms, especially the genetic algorithm, which is crucial in many practical applications [7].

However, it is essential to note that this research has some limitations, especially regarding the specific configuration of our test environment and the characteristics of our data, precisely the use of accurate data and the execution of this distributed version in a real cluster.

Future research could expand on these aspects of this study by exploring other configurations and further evaluating performance in different scenarios. In this context, we can use also our methods implemented in [8–24].

Ultimately, this study can be a starting point for other researchers and students interested in combining evolutionary algorithms and distributed computing. The potential of this approach is immense, and there is still much to discover. We hope this research will stimulate more innovative work.

References

1. Carnein, M., Trautmann, H.: evoStream—evolutionary stream clustering utilizing idle times. Big Data Res. **14**, 101–111 (2018)
2. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Natural Computing Series. Springer Berlin Heidelberg, Berlin, Heidelberg (2003)
3. Maulik, U., Bandyopadhyay, S.: Genetic algorithm-based clustering technique. Pattern Recogn. **33**, 1455–1465 (2000)
4. Xu, D., Tian, Y.: A comprehensive survey of clustering algorithms. Ann. Data Sci. **2**, 165–193 (Jun 2015). Company: Springer Distributor: Springer Institution: Springer Label: Springer Number: 2 Publisher: Springer Berlin Heidelberg
5. Silva, J.A., Faria, E.R., Barros, R.C., Hruschka, E.R., Carvalho, A.C.P.L.F.D., Gama, J.: Data stream clustering: a survey. ACM Comput. Surv. **46**, 1–31 (2013)
6. Hahsler, M., Bolaos, M.: Clustering data streams based on shared density between micro-clusters. IEEE Trans. Knowl. Data Eng. **28**, 1449–1461 (2016)
7. Matloff, N.: Programming on Parallel Machines
8. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
9. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of Things and Cloud Computing (2016)
10. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
11. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv: 1307.5910
12. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and Skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
13. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
14. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
15. Idrissi, A, Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
16. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020).
17. Elhandri, K., Idrissi, A.: Comparative study of Topk based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10**(1-2) (2020)

18. ElHandri, K., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
19. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
20. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
21. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
22. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
23. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
24. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)

Explainable Multi-agent Network for Multi-classification in Small Tabular Data

Mehdi Bouskri and Abdellah Idrissi

Abstract Deep learning has gained tremendous success in recent years in most data types, but still outperformed by classical machine learning on tabular datasets, the most common data type, especially smaller ones. An explainable deep learning model with accurate prediction power on tabular datasets can make it easier to use the advantages of such methods to allow more flexibility in architecture design to be applied on this type of data, the use of transfer learning on similar tasks, and build trust in model predictions. We present an explainable multi-agent network for multi-classification of small tabular datasets, that uses a discrimination network, an attention block, and a classification network. Our model has an accuracy close to the best results of OpenML benchmark on all datasets used with an average difference of 4.2%, and outperformed a similar deep learning approach.

Keywords Neural network · Multi-classification task · Tabular data

1 Introduction

Tabular data being the most common data type with a high potential artificial intelligence value impact [1], still under explored by deep learning models in favor of images, text and audio. Having explainable deep learning models that can perform well on problems with tabular datasets especially when the datasets are small, which is the case in most healthcare problems where additional data points are hard or impossible to create, and where human experts need to be involved in the decision making process, can offer several advantages. These advantages are not limited to flexibility in architecture design, transfer learning of pre-trained models on similar tasks, allowing the possibility of semi-supervised learning for tabular data, facilitat-

M. Bouskri (✉) · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team Faculty of Science of Rabat, Mohammed V University, Rabat, Morocco
e-mail: mehdi_bouskri@um5.ac.ma; m.bouskri@gmail.com

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_5

ing collaboration between machines and human, and help build trust in the model's predictions.

The performance gap between machine learning and deep learning models made evident by Grinsztajn et al. [2], where tree-based models outperformed deep learning models in balanced tabular datasets with samples between 3000 and 10000, and more than 4 features, in speed and accuracy on classification tasks as well as regression tasks. Shwartz-Ziv and Armon [3], reported an inferior performance of deep learning models on several tabular datasets compared to tree-based models. This gap was our motivation to explore this domain of deep learning and try to optimize deep learning performance and give more credibility to its predictions.

In this paper, we propose a multi-agent architecture with attention mechanism trained in parallel and in few-shot learning fashion, that we applied on four small datasets, three from OpenML Benchmark [4] and one from UCI Machine Learning Repository [5]. Our model outperformed TabNet [6] on all datasets and had a close performance compared to the best runs from OpenML Benchmark with an average difference in accuracy of 4.2%. The paper is organized as follows: related work in Sect. 2, network and datasets descriptions in Sect. 3, results and discussion in Sect. 4, and conclusion in Sect. 5.

2 Related Work

Deep learning approaches can tackle tabular data problems either by encoding it into a more informative data type [7], by using hybrid architecture with classical machine learning approaches [8], by using transformers [6] or by using a regularized deep neural network [9]. Despite the high number of explored deep learning approaches, decision tree ensembles still outperform deep learning models on most datasets, especially on small datasets [10].

Our work is inspired by Discrimination Neural Network architecture, a binary classification model based on cosine distance [11], which outperformed OpenML benchmarks on multiple datasets. Since our model is trained in a few-shot learning style, where each input is compared to n pairs to compute the similarity, it's compared to few-shot models that are successful on image and text data such as Matching Networks [12].

We will compare our model performance to TabNet, an interpretable deep tabular data learning transformer-based model with feature selection [6]. Using multiple networks like a decision tree, TabNet aggregates all networks outputs to make predictions. It succeeded on a variety of large tabular datasets compared to other machine learning methods.

3 Network Architecture and Datasets

For each agent in training phase, the input consists of n pairs that includes $x1$, instance that we want to classify, and $x2_i$, instance that will be used to measure the similarity to $x1$, so that the input is $(x1, x2_i)$ with i from 1 to n. This input will be used by the first part of the network called discriminator to create an embedding representation $emb1$ and $emb2_i$ for $x1$ and $x2_i$ respectively. This part of the network uses cosine distance as a loss function. The second part of the network called classifier takes only $emb1$ as input to predict if it belongs to the specific class of the agent or not. Essentially each agent will be trained to differentiate between one class and the rest, meaning that the model will have N agents with N is the number of classes in the dataset (Fig. 1).

All agents are trained in parallel following the same training loop as used in [11], at the start of an episode input $\{x1, x2_i\}_{i=1}^{n}$, similarity label l_i, and class label y_i are extracted. Similarity label is equal to 1 if pairs are from the same class and –1 if not. Class label y_i equals 1 if the class is specific to the agent and 0 if not. An episode has an inner epoch where the discriminator network will compare $x1$ to each $x2_i$, and cosine loss is calculated using similarity labels and embedding outputs. Finally, the classification network uses embedding output of $x1$ to predict the class label $\hat{y}_i$ and classification loss is calculated between y_i and $\hat{y}_i$. The model is optimized using both classifier and discriminator losses.

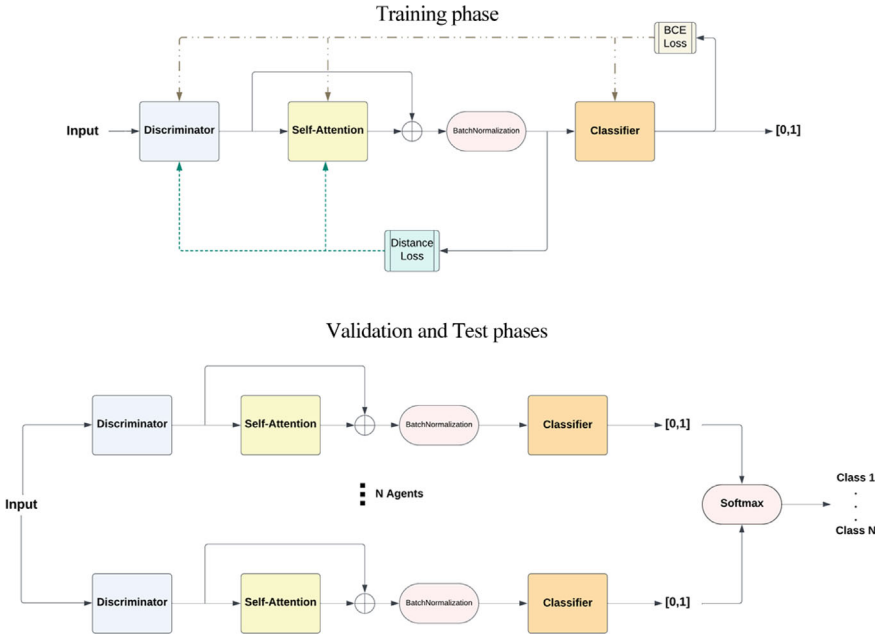


Fig. 1 Model architecture in training, validation and test phases

Table 1 Dataset's characteristics

Dataset	Classes	Features	Instances	Classes weights
53[a]	4	19	846	1.03, 1, 1.03, 0.94
146821[a]	4	7	1728	0.89, 0.16, 2.8, 0.15
146822[a]	7	17	2310	1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0
Land Mines	5	4	338	0.98, 1.03, 0.98, 0.96, 1.05

[a] OpenML task id

For validation and test phases, all agents work jointly with the same input x and output a probability of x being from the agent's class, then a SoftMax function is performed on the joint probability tensor to predict the actual class of the input. For the limited computation constrains we faced during model hyperparameters search, we limited the search only to dropout rate of classifier network, embedding dimensions of the discriminator network, and learning rate. For all dataset we used the same number of layers for all networks, 2 layers for discriminator network, a single head attention, three layers classification network, and $n = 5$ pairs. We used a varying number of batch sizes and layer units in each network depending on dataset. TabNet trained with the same data splits as our model with parameters width of the decision $n_d = 8$ and Width of the attention $n_a = 8$.

We chose four datasets with instances ranging from 338 to 2310, three datasets from OpenML Benchmark and one from UCI Machine Learning Repository. All dataset's characteristics are described in Table 1. Datasets were split by a stratified 10-fold scheme into training set (with 10% for validation set) and test set to make sure that all classes are being represented with the same percentage in all splits.

4 Results and Discussion

As expected, a higher number of instances in a dataset results in a higher performance in general. Our model accuracy is better than TabNets in all datasets, and it's less than the best accuracy in OpenML benchmark by 4.3% for dataset 53, by 3.1% for dataset 146821, and by 5.2% for dataset 146822 (Table 2). Land Mines accuracy wasn't reported to be measured by a cross validation, and our model performs poorly in comparison to the meta-heuristic k-nearest neighbors (k-NN) used for the prediction, due to the low number of points in this dataset.

The average Receiver Operating Characteristic Area Under the Curve (ROC AUC) scores on all datasets for our model was lower than TabNet and OpenML benchmark despite the higher accuracy (Table 3), which can be explained by the fact that our model has varying ROC AUC scores between the classes of the same dataset (Figs. 2 and 3). The varying ROC AUC score by class of our method does not always correlate

Table 2 Accuracy results

Dataset	Baseline	TabNet	Ours
53	0.87	0.732	0.827
146821	1.0	0.944	0.969
146822	0.981	0.886	0.929
Land Mines	0.858[a]	0.526	0.654

[a] Results of a Meta-heuristic k-NN from [13]

Table 3 Average ROC AUC scores on all classes

Dataset	Baseline	TabNet	Ours
53	0.914	0.918	0.886
146821	1.0	0.993	0.945
146822	1.0	0.985	0.959
Land Mines	–	0.804	0.781

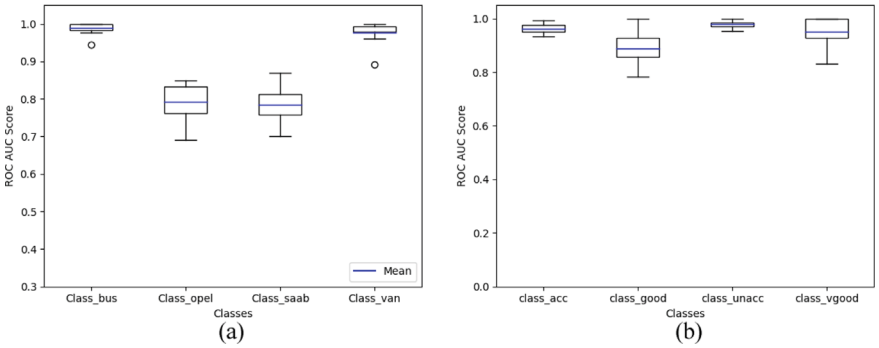


Fig. 2 Our model's ROC AUC scores by class: **a** dataset 53, **b** dataset 146821

with classes weights, and in some cases classes with a high number of important features has a higher ROC AUC score.

Our proposed architecture gives an insight on the importance given to each feature by the attention block in order to justify its prediction. During each phase, attention weights can be extracted for each input. Figures 4 and 5 represent the average attention weights of features by classes, on all correct predictions of the test split for all datasets used.

Our method needs much more computation when the dataset contains a higher number of classes, a high number of features or both. While the high number of features might be overcome by limiting training on only high variance features, a high number of classes can only be addressed by adding more CPUs or even training on multiple GPUs if possible.

Intuitively one might think that a small dataset (in terms of number of features and instances) needs a small network, that was not the case with Land Mines dataset

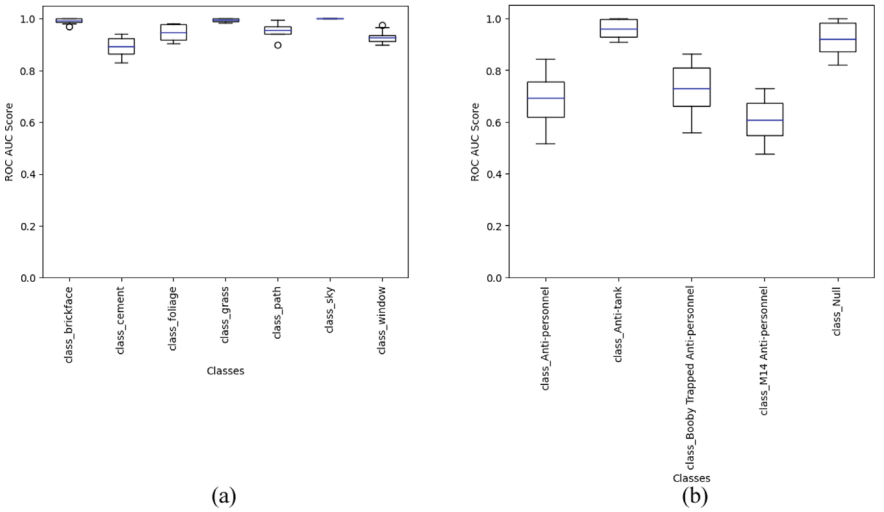


Fig. 3 Our model's ROC AUC scores by class: **a** dataset 146822, **b** Land Mines dataset

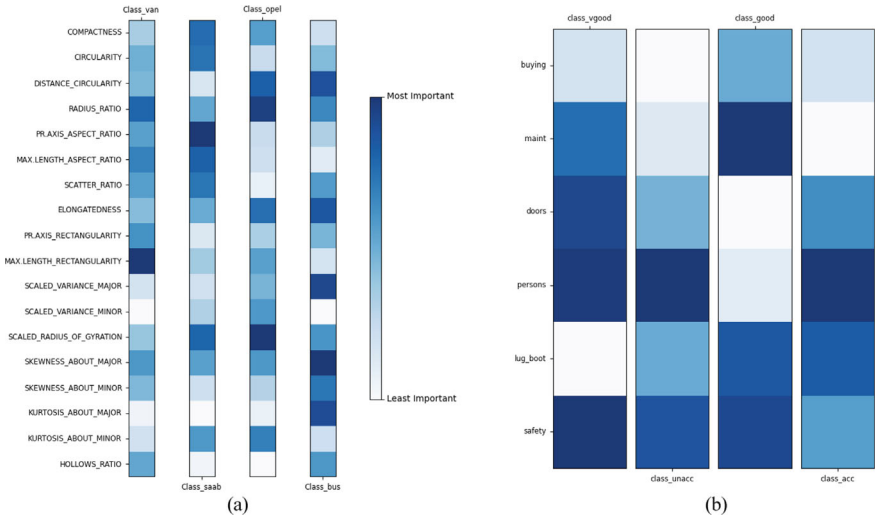


Fig. 4 Our model's attention weights by features: **a** dataset 53, **b** dataset 146821

which contains three features and five classes. The model configuration to train on Land Mines dataset was the biggest of all models used in this paper.

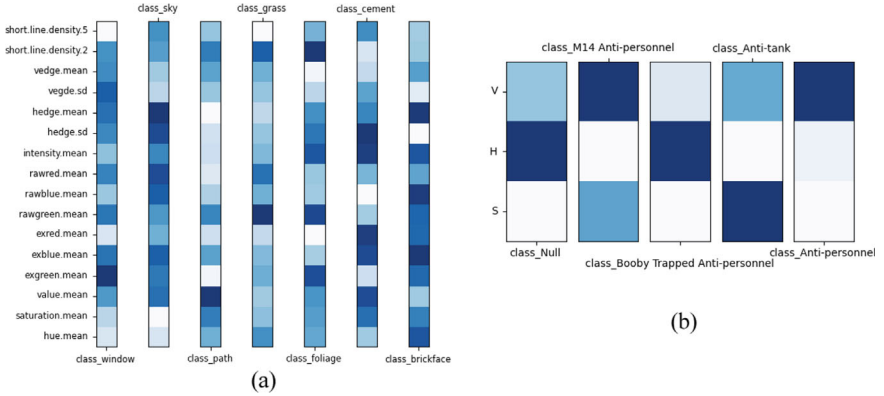


Fig. 5 Our model's attention weights by features: **a** dataset 146822, **b** Land Mines dataset

5 Conclusion

We presented an explainable multi-agent network for multiclass classification of small tabular datasets. Each agent is trained in parallel to differentiate between one class and the rest, and in the test phase all agents are used jointly to predict the input class. While we did not use any features extraction or data augmentation, our model accuracy surpasses TabNet on all datasets, and it's close to the best performances in OpenML benchmark.

Model's performance can benefit from an extended hyperparameters search that includes multiple distance loss functions, since the model is sensitive to hyperparameters choices depending on the used data and one fixed distance loss function might not be adequate to all types of datasets. Further investigation is needed to assess how datasets characteristics affect our model performance, and to select the best initial hyperparameters to minimize the search space, since relying only on the size of datasets does not reflect the real complexity of it. We also intend to introduce these concepts into other themes such as those described in [14–30] for their possible improvements.

References

1. Chui, M., Manyika, J., Miremadi, M., Henke, N., Chung, R., Nel, P., Malhotra, S.: Notes from the AI frontier: insights from hundreds of use cases. McKinsey Glob. Inst. **2** (2018)
2. Grinsztajn, L., Oyallon, E., Varoquaux, G.: Why do tree-based models still outperform deep learning on typical tabular data? Adv. Neural Inf. Process. Syst. **35**, 507–520 (2022)
3. Shwartz-Ziv, R., Armon, A.: Tabular data: deep learning is not all you need. Inf. Fusion **81**, 84–90 (2022)
4. Bischl, B., Casalicchio, G., Feurer, M., Gijsbers, P., Hutter, F., Lang, M., Mantovani, R.G., van Rijn, J.N., Vanschoren, J.: Openml benchmarking suites (2017). arXiv:1708.03731

5. Kahraman, H.T.: Land Mines. UCI Machine Learning Repository (2022). https://doi.org/10.24432/C54C8Z
6. Arik, S.Ö., Pfister, T.: Tabnet: attentive interpretable tabular learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, pp. 6679–6687 (2021)
7. Yoon, J., Zhang, Y., Jordon, J., van der Schaar, M.: Vime: extending the success of self-and semi-supervised learning to tabular domain. Adv. Neural Inf. Process. Syst. **33**, 11033–11043 (2020)
8. Ke, G., Xu, Z., Zhang, J., Bian, J., Liu, T.Y.: DeepGBM: a deep learning framework distilled by GBDT for online prediction tasks. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, pp. 384–394 (2019)
9. Shavitt, I., Segal, E.: Regularization learning networks: deep learning for tabular datasets. Adv. Neural Inf. Process. Syst. **31** (2018)
10. Borisov, V., Leemann, T., Seßler, K., Haug, J., Pawelczyk, M., Kasneci, G.: Deep neural networks and tabular data: a survey. IEEE Trans. Neural Netw. Learn. Syst. (2022)
11. Munkhdalai, L., Munkhdalai, T., Hong, J.E., Pham, V.H., Theera-Umpon, N., Ryu, K.H.: Discrimination neural network model for binary classification tasks on tabular data. IEEE Access **11**, 15404–15418 (2023)
12. Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al.: Matching networks for one shot learning. Adv. Neural Inf. Process. Syst. **29** (2016)
13. Yilmaz, C., Kahraman, H.T., Söyler, S.: Passive mine detection and classification method based on hybrid model. IEEE Access **6**, 47870–47888 (2018)
14. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Arch. **9**(2–3), 136–148 (2020)
15. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular ad-hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
16. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv:1307.5910
17. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
18. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
19. handri, K.E., Idrissi, A.: Comparative study of top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
20. Idrissi, A., Li, C.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
21. Idrissi, A., Yakine., F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
22. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
23. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
24. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless ad hoc networks using the skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of Things and Cloud Computing (2016)
25. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
26. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol., 5567–5584 (2023)

27. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
28. Handri, K.E., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
29. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
30. Elhandri, K., Idrissi, A.: Parallelization of top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)

Agriculture Recommendation System Using Collaborative Filtering

Chahrazad Lagrini and Abdellah Idrissi

Abstract The agricultural sector plays a crucial role in the Moroccan economy, making a significant contribution to the GDP and directly influencing economic growth. However, it faces various challenges despite the efforts of the authorities. In this context, our work aims to propose solutions using artificial intelligence techniques to assist farmers in selecting crops suitable for their land while considering environmental factors. The recommendation system is designed to suggest which crops to grow based on location, soil properties, and weather details operates by analyzing historical data on crop performance across various locations. It uses a method known as collaborative filtering, adapted to the context of agricultural data, to make these recommendations. This recommendation system works by learning from historical data on how different crops perform in various environmental conditions. It then uses this knowledge to predict which crops are likely to be successful in similar conditions elsewhere. This process involves sophisticated data analysis techniques but aims to provide actionable, evidence-based advice to farmers and agricultural planners on optimizing crop selection for any given location.

Keywords Recommendation system · Crop agriculture · Collaborative filtering · Morocco

1 Introduction

The agricultural sector plays a fundamental role in the Moroccan economy, contributing significantly to GDP and having a direct impact on the country's growth rate. However, this essential sector faces various challenges that require innovative

C. Lagrini (✉) · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team, Computer Science Laboratory (LRI), Computer Science Department, Faculty of Science of Rabat, Mohammed V University in Rabat, Rabat, Morocco
e-mail: chahrazad.lagrini@um5r.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_6

solutions. Despite the interventions of the ministry and relevant organizations, there remains a pressing need for proposed solutions to sustainably improve the agricultural situation in Morocco.

From this perspective, our project aims to make a significant contribution to improving Moroccan agriculture by exploiting advances in artificial intelligence. Our main objective is to assist farmers in making crucial decisions regarding crop selection, taking into account the environmental parameters specific to their land.

To achieve this goal, we propose to implement artificial intelligence methods, including machine learning, to create an efficient and accurate recommendation system. This system will be based on the experimentation of machine learning models which will exploit soil data and meteorological characteristics of the field. This research project is therefore intended to be an innovative and holistic approach to improving the Moroccan agricultural sector. By exploiting the possibilities offered by artificial intelligence, we aspire to provide farmers with powerful decision support tools adapted to their specific needs. We anticipate that the establishment of an accurate and efficient recommendation system will help optimize the use of agricultural resources, increase yields and reduce risks associated with crop choices. this project aims to be a step forward towards more intelligent and sustainable agriculture in Morocco. By combining soil and weather data with advances in artificial intelligence, we hope to offer innovative and concrete solutions to meet the challenges of the Moroccan agricultural sector and thus contribute to its long-term growth and prosperity.

2 Agriculture in Morocco

2.1 Agricultural Policies

Agriculture has undeniable economic and social importance in Morocco, with a share of around 38% in total employment at the national level and around 74% in rural areas. This activity also contributes nearly 13% of GDP, knowing that this contribution varies depending on the territory. For some regions, the agricultural sector represents a preponderant part of economic activity.

Since its independence, Morocco has implemented several agricultural policies to develop and modernize the country's agricultural sector. These main policies include:

Green Morocco Plan (2008–2020): Launched in 2008, the Green Morocco Plan was a major initiative aimed at modernizing and developing the Moroccan agricultural sector. This plan focused on crop diversification, adoption of advanced agricultural technologies, improvement of rural infrastructure, promotion of modern irrigation and establishment of agricultural financing mechanisms.

"Generation Green" strategy (2016–2030): This strategy aims to promote sustainable and high value-added agriculture. It emphasizes innovation, agricultural

research, the preservation of natural resources, the valorization of agricultural products and the promotion of exports.

Post-COVID-19 Agricultural Recovery Plan: In response to the effects of the COVID-19 pandemic on the agricultural sector, Morocco launched a recovery plan to support farmers and ensure the continuity of agricultural activities. This plan includes financial support measures, subsidies for agricultural inputs, infrastructure modernization and the promotion of digital agriculture.

Among the various agricultural policies deployed by Morocco, there are several initiatives and projects that have been put in place to improve agricultural productivity and promote environmental sustainability. Among these initiatives, we can cite Statagri, MOSAICC Morocco and CGMS Morocco. Additionally, the Fertimap initiative stands out as an example of an innovative approach in the field of soil management and precision agriculture.

Statagri is an online service developed and exhibited by the Ministry of Agriculture of Morocco. It aims to collect statistical data from agricultural surveys, as well as to consolidate and archive the statistics produced by the ministry. This data includes information such as land area, agricultural production and livestock numbers. Statagri thus provides an essential tool for monitoring and evaluating the performance of the agricultural sector.

2.2 Climate Change

Climate change is having a significant impact on the agricultural sector in Morocco, creating new challenges and requiring adaptation measures. Changes in weather patterns, such as increasing temperatures, decreasing precipitation and irregular seasons, have direct impacts on agricultural production and the availability of natural resources.

Moroccan farmers face increasing difficulties in planning their agricultural activities due to climate uncertainty. Changes in precipitation patterns and prolonged droughts can lead to decreased water availability for irrigation, thereby affecting crop growth. Additionally, higher temperatures can cause heat stress to crops, reducing their yield and quality.

To face these challenges, Morocco has undertaken various actions to promote climate-resilient agriculture. Investments have been made in modern and more efficient irrigation technologies, such as drip irrigation, which enable more efficient use of water. Research is also being carried out to develop drought-resistant crop varieties adapted to changing climatic conditions.

In addition, adaptation strategies have been put in place to diversify agricultural activities and promote sustainable agricultural practices. This includes promoting agroforestry to improve soil and water conservation, using land conservation techniques to reduce erosion, and establishing early warning systems to help farmers make informed decisions in the face of unpredictable weather conditions.

3 The Effect of Soil and Weather on Crop Types

3.1 Soil Properties

Soil properties play an important role in agricultural production. A cultivated plant requires good and favourable soil conditions and environmental situation to give full potential yield. But the lack of some nutrients due to soil properties can lead to a decline in agricultural production. These properties are therefore essential for producing food of good quality and sufficient quantity.

Yield predictions can be made based on soil characteristics, such as phosphorus, nitrogen [1], pH and porosity. However, it should be noted that when predicting yields under standardized pot conditions, emphasis is placed on variables associated with organic matter. To predict field yields under natural conditions, however, it is necessary that additional variables describing the hydraulic regime and the local climate are known [2].

The authors in the article [3] made a comparison between many crop recommender systems that are present in the literature, and as a sort of synthesis. They tried to categorize the soil elements that were used by these studies as input parameters for these systems, among them we have the following:

Soil physical properties: soil texture, soil color, soil type, soil porosity and bulk density.

Soil chemical properties: soil pH, soil salinity, nutrients availability, soil electrical conductivity…

3.2 Weather Data

Weather conditions are of crucial importance to all crops, as is soil. Each plant has specific requirements for the climate in which it grows, which results in various climatic needs such as solar radiation, temperature and water. These climatic elements can act as limiting factors for agricultural production, whether due to excess or lack.

4 Recommendation Systems

The beginnings of recommendation systems stem from research carried out in the construction of models representing user choices. This research comes from distinct fields such as documentary research, management and marketing sciences, cognitive sciences and approximation theories [4]. Recommendation can be compared to a dialogue between a person who is an expert in a field and another person wishing to acquire information in this field. More concretely, a librarian will be able, depending

on the tastes of one of his clients, to offer a list of works to the latter which will be none other than a recommendation within the meaning of recommendation systems. Considering this analogy, the librarian, given his knowledge of the different works he offers can be seen as the knowledge base of the items to be recommended. It thus knows the items individually and is capable of making item associations according to different criteria characterized by the profile of a user. The latter is in fact aware of his tastes and can submit them to the librarian. We can even go further by assuming that the librarian knows the tastes of different patrons. He would then be able to offer works to a client, which were liked by other similar clients.

The notion of recommendation induced by this example is undoubtedly the basis of the principles of the recommendation system.

This vision of recommendation reinforces the definition of it as being a means of helping a user to choose a certain number of items from a set of resources where he does not have enough knowledge of the latter in order to sort and to evaluate the relevance of the available documents.

Among the first proposed recommendation system approaches, we can cite three pioneering works: Tapestry, GroupLens and Ringo. These approaches laid the foundation for modern recommender systems and contributed significantly to their development.

One such approach is Tapestry, introduced by Goldberg et al. [5]. Tapestry was a content-based recommendation system that used user profiles and item descriptions to generate personalized recommendations. It focused on recommending new information resources in the field of information retrieval. The system used collaborative filtering techniques to find similarities between user preferences and item characteristics, in order to recommend relevant resources.

Another notable approach is GroupLens, developed by Resnick et al. [6]. GroupLens was a recommendation system based on collaborative filtering that aimed to recommend news articles to a group of users. The system collected user ratings on news articles and used this information to find correlations between user preferences and generate personalized recommendations for each member of the group.

Ringo, proposed by Shardanand and Maes [7], was another content-based recommender system. It focused on music recommendation by analysing user-provided text annotations for songs. The system identified similarities between annotations and used this information to generate personalized recommendations for each user.

These early approaches paved the way for much subsequent research in the area of recommender systems. They laid the theoretical and technical foundations for the development of new recommendation methods and algorithms. Over the years, many other approaches have been proposed, leveraging different techniques and algorithmic approaches to improve the accuracy and relevance of recommendations.

In the following sections, we will explore these and other modern recommender system approaches in more detail, examining how they work, their benefits, and their limitations. We will also highlight recent advances and current trends in the field of recommender systems.

In this work, we chose the technique of collaborative filtering.

The notion of collaborative filtering is the basis of the recommendation, content filtering methods being rather linked to so-called personalized information retrieval systems. It is no longer based on the notion of proximity of a pair "new item - user profile" but seeks to bring the current user closer to a set of existing users. The idea here is no longer to focus specifically on the new item that would likely please the user but to look at which items were appreciated by users close to the current user.

Collaborative recommendation systems have the following advantages:

- Use the scores of other users to evaluate the usefulness of the elements;
- To find users or groups of users whose interests correspond to the current user;
- And the more users there are, the more scores there are: the better the results.

However, such systems also have disadvantages:

- Finding similar users or groups of users is difficult;
- The recommendation system comes up against the low density of the Users X Elements matrix;
- Additionally, there is also the cold-start problem: when a new user uses the system, their preferences are not known and when a new item is added to the catalogue, no one has assigned it score;
- In systems with a large number of elements and users, the calculation grows linearly; appropriate algorithms are therefore necessary;
- "Non-diversity": it is not useful to recommend all the films with the actor Al Pacino to a user who liked one of them in the past.

5 Methodology

5.1 Data Collection

Gathering real data relating to the study territory, namely Morocco, represents a complex challenge for this study.

In order to build a database that meets the needs of the project, we undertook both national and international research to collect data on soil, weather and agricultural production in Morocco. We also carried out a first experiment using data from Europe. This process allowed us to develop a data assembly plan divided into four main steps.

The first step consists of acquiring pedological data, that is to say data relating to soil characteristics. Then we proceed to the second step which consists of collecting agricultural production data. Finally, we focus on weather data research. During these three steps, we had to deal with the geolocation constraint, because it was essential to have data linked to specific locations to be able to combine them later.

The fourth and final step of the process consists of gathering data from the different provinces and municipalities of the country. This step is crucial to obtain a complete and representative view of the entire territory.

In summary, our approach to building the database involved extensive research in national and international sources, as well as a four-step assembly process: acquisition of soil data, collection of agricultural production data, search for meteorological data and collection of provincial and municipal data.

Once the data has been collected from the different resources, we find ourselves with this list of data files waiting to be collected in the appropriate way:

- Raw material values for all community points
- Potassium values for all community points
- Phosphorus values for all community points
- PH values for all community points
- Agricultural production in Morocco according to municipalities
- Provinces of Morocco with parents' IDs
- Municipalities of Morocco with parents' IDs.

5.2 Data Preprocessing

The system starts with different datasets which are consolidated into one dataset that contains information about crop production in different locations, along with soil properties (like pH, nitrogen, phosphorus, and potassium levels) and weather conditions (such as temperature, precipitation, and humidity). The data is cleaned and organized so that each row represents a unique combination of a location and a crop, with the production value indicating how well the crop performed in that location.

5.3 Creating a Performance Matrix

The cleaned data is then used to create a "performance matrix." In this matrix, each row represents a different location, and each column represents a different type of crop. The values in the matrix show the production levels of each crop in each location. If a certain crop wasn't grown or data is missing for a particular location, that value is filled with zero, indicating no production.

5.4 Applying Matrix Factorization

Next, the system uses a technique called matrix factorization to simplify and condense the information in the performance matrix. This technique breaks down the large, complex matrix into smaller, more manageable pieces that capture the underlying patterns in the data. These patterns might relate to how certain types of soil, weather conditions, or other factors influence crop production.

5.5 *Making Recommendations*

To recommend which crops to grow in a new or existing location, the system looks at the simplified data from the matrix factorization step. It finds locations that are like the target location based on the conditions and characteristics captured in the data. Then, it looks at which crops perform well in those similar locations and uses this information to make recommendations.

For example, if the target location has soil and weather conditions like another location where barley and wheat thrive, the system might recommend barley and wheat as good choices for the target location (see Figs. 1 and 2).

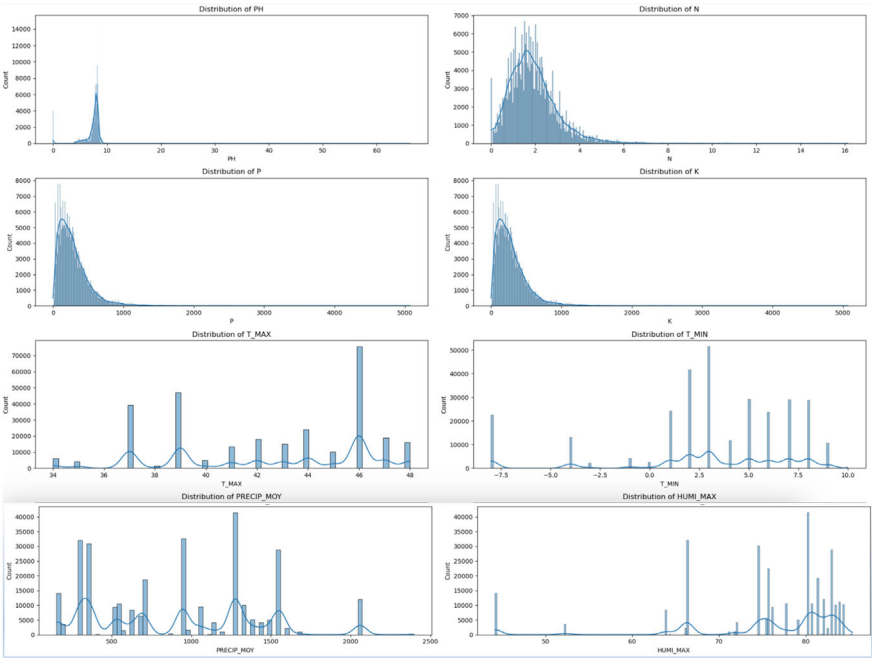


Fig. 1 Plotting distributions of numerical features

```
recommended_crops = recommend_crop("Al Hoceima", "Ajdir", crop_performance_matrix_filled, location_crop_matrix_reduced, top_n=3)
recommended_crops
```

```
culture
orge          910.3
ble tendre    412.7
ble dur       238.4
dtype: float64
```

Fig. 2 Testing the function with an example of a province name and commune name

6 Conclusion

The data structure targeted for this experiment, which consists of data sets containing the chemical and/or physical characteristics of the soil, meteorological parameters as well as crops already cultivated in the same area, with a fairly sufficient level of granularity, is not available to the public. We had a big challenge for data collection and preprocessing.

Basically, our recommendation system works in a way that it first sees what is the location and what is the soil type and the weather type of that particular location. It tries to figure out what are the similar locations where we have same or a close soil type and weather, then it tries to gather all of these similar regions and finds which crop has been more productive at those sides and recommend it. So, it's a perfect system in that scenario where it looks for the most similar and most productive crop. Therefore, we couldn't find an appropriate evaluation to it. Different approaches [8–23] could be adapted in this field.

References

1. Swami, S.: Effect of Soil Biological Properties on Crop Production (2019)
2. Stenberg, B.: Soil attributes as predictors of crop production under standardized conditions. Biol. Fertil. Soils **27**, 104–112 (1998). https://doi.org/10.1007/s003740050407
3. Ommane, Y., Rhanbouri, M., Chouikh, H., Jbene, M., Chairi, I., Lachgar, M., Benjelloun, S.: Université Mohammed VI Polytechnique "Machine Learning based Recommender Systems for Crop Selection: A Systematic Literature Review"
4. Adomavicius, G., Sankaranarayanan, R., Sen, S., Tuzhilin, A.: Incorporating contextual information in recommender systems using a multidimensional approach. ACM Trans. Inf. Syst. **23**, 103–145 (2005)
5. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Commun. ACM **35**, 61–70 (1992)
6. Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., Riedl, J.: GroupLens: An Open Architecture for Collaborative Filtering of Netnews, pp. 175–186. ACM Press (1994)
7. Shardanand, U., Maes, P.: Social Information Filtering: Algorithms for Automating "Word of Mouth", pp. 210–217. ACM Press (1995)
8. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
9. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
10. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv: 1307.5910
11. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
12. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M. : Top-k and Skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
13. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)

14. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
15. Elhandri, K., Idrissi, A.: Comparative study of Top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
16. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
17. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
18. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
19. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
20. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
21. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
22. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
23. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)

A Hybrid Ensemble Approach Integrating Machine Learning and Deep Learning with Sentence Embeddings for Webpage Content Classification

Kerkri Abdelmounaim and Mohamed Amine Madani

Abstract With the continuous increase in online information, the automatic classification of diverse and complex webpage content has become essential. In our study, we propose a novel ensemble learning method that incorporates six different sentence embedding techniques, which are trained using both traditional and deep learning models. Our method utilizes a stacking technique to calculate prediction probabilities. To test the efficacy of our approach, we compare it with two well-known ensemble methods: XGBoost and Random Forest. The comparison includes testing the models on each sentence embedding separately, and jointly through the combination of all embeddings in one dataset. We also compare our method with ensemble techniques like probability averaging and majority voting on both XGBoost and Random Forest. Our method obtained an improvement in classification metrics when compared to both XGBoost and Random Forest, on a scraped dataset containing roughly 15,647 labeled web pages.

1 Introduction

Word embedding refers to the conversion of text into numerical vectors for processing by machine learning models. This step is crucial in all natural language processing (NLP) tasks, as it allows the incorporation of contextual information into the learning process. Unlike simple vectorization, embedding typically involves more sophisticated methods, such as those found in deep learning models. Our study focuses on sentence embeddings, which, while similar to word embeddings, function at the sentence level, offering a more holistic representation of textual meaning. Instead of

K. Abdelmounaim (✉)
Laboratory of Stochastic and Deterministic Modeling National School of Applied Sciences Mohammed Premier University, Oujda, Morocco
e-mail: a.kerkri@umpa.ac.ma

M. A. Madani
Engineering Sciences Laboratory LSI National School of Applied Sciences Mohammed Premier University, Oujda, Morocco
e-mail: m.madani@ump.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_7

converting each word separately, we convert entire sentences into high-dimensional vectors, which are then suitable for analysis by advanced deep learning models and architectures, including ensemble methods. In machine learning, ensemble methods combine multiple models to improve the overall performance and robustness of predictive analytics. Specifically in NLP, the use of ensemble methods aims to leverage the strengths of various individual models to achieve better results in tasks such as text classification, sentiment analysis, and named entity recognition.

It has been demonstrated that ensemble methods can be significantly more efficient than traditional machine learning algorithms in natural language processing (NLP) tasks, such as text classification [1]. Ensemble classifiers, which combine base models like Naïve Bayes, SVM, and logistic regression, have been shown to surpass both standalone and majority-vote classifiers in sentiment classification tasks [2]. Another study explored the combination of convolutional neural networks (CNNs) and long short-term memory (LSTM) models within an ensemble framework for sentiment analysis in Arabic, resulting in improved F1 scores on the standard ASTD dataset [3]. Furthermore, research into the enhancement of text classification accuracy through the integration of ensemble learning, deep learning, and effective document representation methods has proven successful [4].

Most of the studies referenced above employ basic vectorization methods, including bag-of-words [2] and one-hot encoding [1]. While embedding techniques have also been incorporated in ensemble methods utilizing models such as XGBoost and SVM, these have predominantly focused on the word level [4]. This observation highlights a gap in research: the limited exploration of sentence embeddings within an ensemble method framework, which serves as the motivation for this work. In this paper, we introduce a novel ensemble method that integrates six sentence embedding techniques: BERT sentence embeddings [5], GloVe (Global Vectors for Word Representation) sentence embeddings [6], Universal Sentence Encoder [7], spaCy embeddings [8], FastText embeddings [9], and GPT-2 embeddings [10], each derived from a distinct model.

Our approach encompasses a variety of machine learning algorithms, including traditional ones like KNN and Naïve Bayes, as well as deep learning models such as fully connected neural networks and Bi-directional Long Short-Term Memory networks (BiLSTMs) [11], and the well-known ensemble model, XGBoost [12]. After training the embeddings, we aggregate their predictions and subsequently train them using a Decision Tree classifier to generate the final predictions of our method. To comprehensively evaluate the performance of our proposed approach, we conduct comparisons with XGBoost and Random Forest in three distinct scenarios. The first scenario involves training each sentence embedding separately with both XGBoost and Random Forest. The second scenario concatenates all embeddings into a single dataset, on which we train XGBoost and Random Forest. In the last scenario we use probability averaging and majority voting as two distinct ensemble methods on all of the embeddings, once with XGBoost and once with Random Forest. Our proposed method demonstrated superior classification metrics compared to both XGBoost and Random Forest, specifically in the application of sentence embeddings.

2 Overview of Sentence Embedding Techniques

2.1 BERT Sentence Embeddings

BERT is short for Bidirectional Encoder Representations from Transformers, it is a pre-trained model built on the revolutionary architecture of transformers and self-attention mechanisms [13]. The self-attention mechanism is a deep neural network that mimics the human brain's ability to focus on important parts of a sentence in order to derive the overall meaning.

BERT is available in two variants: the base model with 12 transformer encoders and the larger version with 24. This large language model (LLM) has been trained on two main tasks: masked language modeling and next sentence prediction. BERT initially learns embeddings for individual tokens but can also be used to generate sentence embeddings as a unified vector representation for an entire sentence. Comparative studies between BERT's text classification capabilities and traditional machine learning models consistently show BERT's superiority in most NLP tasks

2.2 Universal Sentence Encoder

Universal Sentence Encoder (USE) is a deep neural network developed by Google, generates embeddings solely for sentences and not for individual tokens. USE's generated embeddings can be effectively utilized in various applications such as sentiment analysis, text classification, and measuring sentence similarity.

USE incorporates two encoder models: one based on transformer architecture and another utilizing Deep Averaging Networks (DAN). The process of embedding generation in both models involves several steps. As a first step, sentences are tokenized and converted to lowercase. Depending on the chosen architecture, these sentences are then transformed into 512-dimensional vectors. Finally, these embeddings are fed into a neural network tailored to the specific natural language processing (NLP) task at hand.

2.3 Global Vectors for Word Representation (GloVe)

GloVe, short for Global Vectors for Word Representation, is a probabilistic model that creates embeddings at the word level. This model is based on a count-based framework, emphasizing the global co-occurrence statistics of words, specifically how frequently two words appear together within a sentence. The main principle of GloVe is that the co-occurrence of words provides crucial statistical information for the model to effectively learn word representations. The original paper on GloVe describes its loss function as follows:

$$J = \sum_{i,j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log(X_{ij}))^2 \quad (1)$$

here, V represents the vocabulary size, w_i are the word vectors, $\tilde{w}_j$ are separate context word vectors, and X_{ij} denotes the frequency with which words i and j co-occur. Sentence embeddings can be derived using the averaging of word embeddings within a specific sentence.

2.4 SpaCy Sentence Embeddings

SpaCy's embeddings are based on the 'en_core_web_lg' model, which utilizes a convolutional neural network (CNN) architecture. The model was trained on the extensive OntoNotes dataset, providing 300-dimensional word vectors from over a million unique words. Using these words, the sentence embeddings are computed with a simple averaging approach to capture the semantic essence of sentences.

2.5 FasText Sentence Embeddings

The FastText model for word embeddings, developed by Facebook's AI Research lab, represents words not only as a whole but also as a bag of characters of length n. This unique approach allows the inclusion of out-of-vocabulary words by breaking them down into recognizable n-grams, which is particularly suitable for languages with complex morphologies. FastText, built with hierarchical softmax, significantly optimizes the speed of training, making it a state-of-the-art choice for large-scale text processing tasks. In our proposed method, the retrieved word embeddings of FastText are averaged to calculate sentence-level embeddings.

2.6 GPT-2 Sentence Embeddings

GPT-2, a transformer-based model developed by OpenAI, features a sophisticated transformer-based architecture. Its embeddings are contextually and semantically rich, enabling them to understand and incorporate the context of an entire sentence. Unlike traditional word embeddings, GPT-2 embeddings are dynamic, changing based on adjacent words. The use of mean pooling to aggregate token-level vectors into a single sentence-level vector effectively harnesses the power of GPT-2 embeddings.

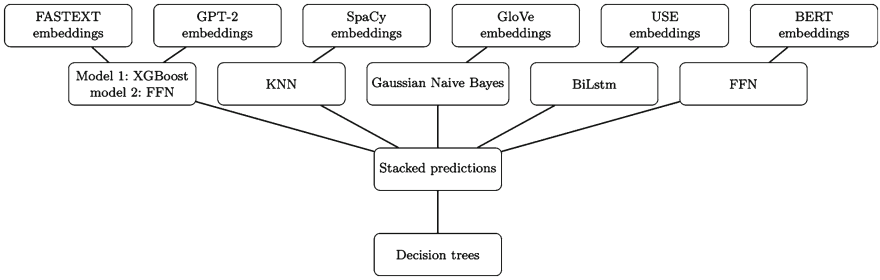


Fig. 1 Overview of the model training architecture

3 Model and Underlying Architectures

Our proposed method leverages a variety of sentence embeddings and machine learning models to encapsulate as much variety as possible. The ensemble method is carried out in two major steps, as shown in Fig. 1.

3.1 Step 1: Training of the Sentence Embeddings Using Base Classifiers

In this section we will only present the architectures of our classifiers, presenting a theoretical background for these models is beyond the scope of this work. The dataset was split into 80% training set and 20% testing set. Here are the details of our base classifiers:

- BERT sentence embeddings are calculated and trained using a Feedforward Neural Network (FFN). In this phase, the training architecture includes two main layers: the first is a dense layer with 128 neurons featuring a ReLU activation function, tailored to handle the feature-rich output of BERT embeddings. The second dense layer has four neurons, corresponding to the number of unique classes in our dataset. A SoftMax activation function is included for effective multi-class classification. Training is conducted over 10 epochs with a batch size of 32.
- The Universal Sentence Encoder (USE) embeddings are trained using Bidirectional Long Short-Term Memory (BiLSTM) networks, known for their capacity to capture sequential data dependencies. The BiLSTM layer has 64 units, followed by a dense layer with a SoftMax activation function, similar to the previous architecture. The model is trained over 10 epochs with a batch size of 32, incorporating a 10% validation split (10% taken from the training set).
- GloVe and SpaCy sentence embeddings are trained using Gaussian Naïve Bayes and K-Nearest Neighbor (KNN) respectively. Gaussian Naive Bayes, a probabilistic classifier, assumes independence of features and, despite its simplicity, has

proven effective in most NLP tasks. The KNN algorithm assigns a label to a new observation based on the majority class among its nearest neighbors. We deliberately included these traditional classifiers to ensure a diverse training process.

- FastText and GPT-2 have been trained twice, once using a three-layered Feed-forward Neural Network (FNN), and once using the XGBoost model. The first layer of the FNN for GPT-2 embeddings has 512 neurons with a ReLU activation function; this number of neurons corresponds to the feature space dimension of the embeddings. The two hidden layers have 256 and 128 neurons, respectively, with the same activation function as the initial layer. To speed up training and mitigate sensitivity to network initiation, we used batch normalization. A dropout rate of 0.5, 0.4, and 0.3 was applied to each of the three dense layers, respectively, to prevent overfitting. In addition to the previous layers, a final layer with four neurons is included, featuring a Softmax activation function for predictive purposes.

3.2 Step 2: Meta-Classification Using Decision Trees

In the meta-classification step, we aggregate predictions from all our base classifiers to make a final prediction. This technique is often referred to as stacking. The goal is to obtain a new set of features, known as meta-features, which are the predictions of each base classifier on the test data. For classifiers that output probability distributions, such as BERT-FNN and USE-BiLSTM, we use the class with the maximum likelihood of occurrence. For other classifiers like KNN, Gaussian Naive Bayes, and XGBoost, we use the predictions as they are. The FNN and XGBoost predictions are utilized more than once to give these models additional weight in the final decision-making process.

4 The Comparative Analysis Scheme

To evaluate our proposed method, we explored the use of chosen sentence embeddings in conjunction with two widely-used ensemble learning algorithms: XGBoost and Random Forest. In addition to comparing our ensemble method with these models, we also aim to assess how different embeddings impact the performance of these algorithms in text classification tasks.

The two models, XGBoost and Random Forest, have been trained with each type of sentence embedding on 80% of our dataset. The choice of XGBoost was motivated by its efficiency and effectiveness in various machine learning contexts, including NLP applications, primarily due to its robust gradient boosting framework. Random Forest, an ensemble of decision trees, was selected for its robustness and ability to prevent overfitting, which is crucial in handling high-dimensional data like sentence embeddings.

The second phase of our comparative analysis involves training XGBoost and Random Forest using two distinct approaches: probability averaging and majority voting.

- **Majority Voting**: In this step, the test dataset was used to collect the class predictions from each of the six trained models for both XGBoost and Random Forest. The majority voting mechanism was then applied to these predictions. The final predicted class for each sample in the test dataset was determined by the mode of predictions across the models. This majority voting process assumes that the most common prediction among the different models is likely to be accurate. The majority voting strategy is particularly effective in reducing the impact of any single model's biases or errors.
- **Probability Averaging**: In this method, instead of using the most likely predictions, we calculated the average of the class probabilities from each sentence embedding for both Random Forest and XGBoost classifiers. The goal is to smooth out extreme predictions and reduce overfitting. After probability averaging, we selected the class with the highest average probability as the final prediction for each sample in the dataset.

In the last step, we combined all six sentence embeddings into a single dataset, aiming to capture a more comprehensive representation of the textual data. The concatenated dataset was then trained using both XGBoost and Random Forest.

5 Results

5.1 Dataset

In our research, we employed a dataset with 15,647 web pages from a news website (80% for training, and 20% for testing), categorized into four distinct topics: Coronavirus, Economy, Africa, and Sports. This dataset was specifically chosen to encompass a wide variety of topics, ensuring a comprehensive base for our text classification study. The dataset is relatively balanced, with the Coronavirus and Economy categories being the largest, comprising 5,008 and 4,992 articles, respectively.

The remaining categories, Africa and Sports, with 3,305 and 2,342 articles, respectively, added further diversity to the dataset. This range of topics allowed us to test the adaptability and accuracy of our models across different content domains.

Table 1 Performance metrics of XGBoost models using different sentence embeddings

Embedding type	Accuracy (%)	Precision	Recall	F1-Score
BERT	77.00	0.79	0.78	0.78
spaCy	78.77	0.80	0.79	0.79
USE	80.00	0.81	0.81	0.81
GloVe	78.77	0.80	0.79	0.80
FastText	78.00	0.80	0.79	0.79
GPT-2	81.98	0.83	0.82	0.83

5.2 Classification Metrics for Individual Sentence Embeddings

5.2.1 XGBoost Results

The performance metrics of XGBoost models, trained on the six sentence embeddings and summarized in Table 1, reveal interesting insights into the effectiveness of each embedding type. GPT-2 embeddings demonstrated the highest accuracy and F1-score, indicating a strong ability to capture contextual information, which is crucial for classification tasks. USE embeddings also showed strong performance, attributable to their direct learning of sentence embeddings rather than merely averaging them from word embeddings.

GloVe, FastText, and spaCy embeddings exhibited comparable results, suggesting their robustness in feature representation for the XGBoost algorithm. Although BERT embeddings slightly lagged in accuracy, their precision and recall scores were competitive, highlighting their potential in text classification.

5.2.2 Random Forest Results

Analyzing the performance of Random Forest models across the various sentence embeddings, as detailed in Table 2, provides valuable insights into how each embedding influences the model's effectiveness in text classification tasks. Once again, GPT-2 stands out with the highest scores across all metrics. This might be attributed to Random Forest's ability to handle the complex decision boundaries introduced by GPT-2's embeddings.

The results for BERT, spaCy, GloVe, and FastText exhibit a degree of parity, indicating that Random Forest models can maintain stable performance across diverse sentence embeddings, from the deeply contextualized to the more traditional word vector approaches.

When compared to the Random Forest models, XGBoost may offer a slight edge, particularly with embeddings like GPT-2, which could benefit from the sequential correction of errors provided by the gradient boosting technique. Nonetheless, the

Table 2 Performance metrics of random forest models using different sentence embeddings

Embedding type	Accuracy (%)	Precision	Recall	F1-Score
BERT	75.46	0.77	0.76	0.76
spaCy	75.71	0.77	0.76	0.76
USE	78.43	0.80	0.79	0.79
GloVe	76.06	0.78	0.77	0.77
FastText	75.49	0.77	0.76	0.77
GPT-2	79.35	0.81	0.80	0.80

Random Forest models still demonstrate strong and competitive performance, making them a reliable choice for text classification tasks where interpretability and variance reduction are key considerations.

5.3 Classification Metrics for the Combined Sentence Embeddings

In this section, we will present the results of XGBoost and Random Forest on the dataset containing all combined sentence embeddings.

5.3.1 Random Forest

The Random Forest model was tuned using a grid search across twelve different variations. It performed well on the combined embeddings, reaching an overall accuracy of 80.63% across 3,129 samples.

The model demonstrated strong precision in the 'Africa' and 'Sports' categories, both scoring 0.88, as detailed in Table 3. 'Sports' achieved the highest recall at 0.88, indicating its effectiveness in capturing relevant cases. The F1-scores were relatively high, with 'Sports' achieving the highest value of 0.88, indicating a good balance between precision and recall.

Overall, the model's recall and F1-score averaged out to 0.81, demonstrating consistent performance across different topics. This suggests that combining embeddings is an effective strategy for text classification tasks.

5.3.2 XGBoost

Following the same methodology as with Random Forest, we performed a grid search for hyperparameter tuning on the XGBoost model. This involved two cross-validation folds for each of the four parameter sets, totaling eight fitting steps. When trained

Table 3 Random forest tuned performance metrics for combined embeddings

Category	Accuracy (%)	Precision	Recall	F1-Score
Africa		0.88	0.73	0.80
Coronavirus		0.79	0.77	0.78
Economy		0.75	0.86	0.80
Sports		0.88	0.88	0.88
Macro avg.	80.63	0.82	0.81	0.81

Table 4 XGBoost tuned performance metrics for combined embeddings

Category	Accuracy (%)	Precision	Recall	F1-Score
Africa		0.84	0.78	0.81
Coronavirus		0.80	0.78	0.79
Economy		0.79	0.84	0.81
Sports		0.87	0.87	0.87
Macro avg.	81.37	0.82	0.82	0.82

on the combined embeddings, the XGBoost model yielded an overall accuracy of 81.37% (Table 4), indicating a high level of predictive performance in comparison to the previously mentioned Random Forest model.

The model's precision and recall were particularly strong in the 'Sports' category, scoring 0.87 for both. The 'Africa' category followed closely with a precision of 0.84 and a recall of 0.78, suggesting a higher rate of false negatives. Categories 'Coronavirus' and 'Economy' displayed balanced precision and recall, each contributing to a stable F1-score of approximately 0.80. The macro average precision and recall across all categories stood at 0.82, with a corresponding F1-score of 0.82, indicating consistent performance across different types of text data.

5.4 Majority Voting and Probability Averaging Approaches

For these two ensemble learning methods, we utilized the XGBoost and Random Forest algorithms. With the Random Forest algorithm, the Majority Vote method exhibited an accuracy of 79% and F1-score of 80% (Table 5), with both precision and recall at 0.80. Small improvements were observed with the Average Probabilities method, where precision marginally increased to 0.81, while recall slightly decreased to 0.79, maintaining an F1-score of 0.80.

In the case of the XGBoost algorithm, the Majority Vote method achieved a higher accuracy of 80.89% and an F1-score of 0.82, complemented by precision and recall scores of 0.82 and 0.81, respectively. The Average Probabilities method further

Table 5 Performance comparison of ensemble methods with XGBoost and random forest

Algorithm	Ensemble method	Accuracy (%)	Precision	Recall	F1-Score
Random forest	Majority Vote	79.00	0.80	0.80	0.80
	Average probabilities	79.03	0.81	0.79	0.80
XGBoost	Majority vote	80.89	0.82	0.81	0.82
	Average probabilities	81.53	0.83	0.82	0.82

enhanced performance, resulting in an accuracy of 81.53% and an F1-score of 0.82, with precision peaking at 0.83 and recall at 0.82.

The results clearly indicate that both algorithms benefit from the Average Probabilities method, with XGBoost showing superior performance in all evaluated metrics. The nuanced improvement across metrics suggests that averaging probabilities may be more effective at capturing predictive certainty across models, leading to a more accurate ensemble.

5.5 Classification Metrics of the New Ensemble Method

The performance of our ensemble method across the different news categories is reported in Table 6. In the 'Coronavirus' category, the model achieved a precision of 0.94 and a recall of 0.90, indicating high performance in its predictions. For the 'Economy' category, precision and recall were recorded at 0.89 and 0.88, respectively.

In the 'Africa' category, the precision and recall were 0.90 and 0.91, respectively, manifesting reliable and consistent classification ability. Impressively, the 'Sports' category saw exceptional results, with a precision of 0.91 and a recall of 0.97, indicating the model's particular adeptness at classifying sports-related content.

Overall, the model exhibited an accuracy of 0.91 across the 3,129 articles in the test set, underlining its robustness. These results highlight the success of our ensemble approach in addressing the challenges of multi-category text classification. The evidence suggests that our proposed ensemble method outperforms the models and methodologies we applied in our comparative analysis. This includes individual sentence embeddings trained on both XGBoost and Random Forest, concatenated embeddings, and the averaging and voting techniques employed with these algorithms. The results establish our method as a highly effective tool for news categorization tasks within the context of our study.

Table 6 Classification metrics for the novel ensemble method

Label	Precision	Recall	F1-Score
Coronavirus	0.94	0.90	0.92
Economy	0.89	0.88	0.88
Africa	0.90	0.91	0.91
Sports	0.91	0.97	0.94
Accuracy	0.91		

6 Conclusion

In this study, we proposed a method that integrates six sentence embedding techniques into an ensemble learning framework for text classification tasks. This method was tested on a real dataset in a comparative analysis, which included two traditional ensemble methods: XGBoost and Random Forest, used with sentence embeddings. Our method has shown promising results, outperforming both models in several scenarios, as demonstrated in the results section.

Our findings suggest that leveraging diverse sentence embeddings can capture complex textual features more effectively than traditional single-model approaches. This is particularly true when these embeddings are trained on a variety of traditional and deep learning models. Further investigation is warranted to explore the method's performance across various datasets and to optimize the computational efficiency of embedding calculations.

Future research should aim to address these perspectives, perhaps by testing the method on multiple datasets and investigating approaches to reduce the computational load of sentence embeddings. This would enable the scaling of calculations in a more seamless manner.

References

1. Mohammed, A., Kora, R.: An effective ensemble deep learning framework for text classification. J. King Saud Univ. Comput. Inf. Sci. **34**(10), 8825–8837 (2022)
2. Saleena, N.: An ensemble classification system for twitter sentiment analysis. Procedia Comput. Sci. **132**, 937–946 (2018)
3. Heikal, M., Torki, M., El-Makky, N.: Sentiment analysis of Arabic tweets using deep learning. Procedia Comput. Sci. **142**, 114–122 (2018)
4. Kilimci, Z.H., Akyokus, S.: Deep learning-and word embedding-based heterogeneous classifier ensembles for text classification. Complexity (2018)
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding (2018). arXiv:1810.04805
6. Pennington, J., Socher, R., Manning, C.D.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543 (Oct 2014)

7. Cer, D., Yang, Y., Kong, S.Y., Hua, N., Limtiaco, N., John, R.S., Kurzweil, R.: Universal sentence encoder (2018). arXiv:1803.11175
8. Honnibal, M., Montani, I., Van Landeghem, S., Boyd, A.: spaCy: industrial-strength natural language processing in python (2020)
9. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. Trans. Assoc. Comput. Linguist. **5**, 135–146 (2017)
10. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language models are unsupervised multitask learners. OpenAI Blog **1**(8), 9 (2019)
11. Graves, A., Fernández, S., Schmidhuber, J.: *Bidirectional LSTM networks for improved phoneme classification and recognition*. In: International Conference on Artificial Neural Networks, pp. 799–804. Springer Berlin Heidelberg, Berlin, Heidelberg (Sep 2005)
12. Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794 (Aug 2016)
13. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Polosukhin, I., et al.: Attention is all you need. Adv. Neural Inf. Process. Syst. **30** (2017)

Creating a Customized Dataset for Financial Pattern Recognition in Deep Learning

Mustapha El Bakai, Youness Boutyour, and Abdellah Idrissi

Abstract In the evolving domain of financial markets, the precision of pattern recognition significantly influences investment decisions. This paper presents a novel approach to creating and evaluating custom datasets tailored for deep reinforcement learning (DRL) applications in stock chart pattern recognition. Addressing the gap in standardized dataset preparation, we meticulously develop datasets, focusing on popular patterns like Double Tops and Double Bottoms. Utilizing line charts, we emphasize the importance of dataset quality over quantity, a paradigm shift from conventional candlestick reliance. Our methodology entails a multi-faceted process involving image collection, annotation, and segmentation, ensuring robustness and diversity. We explore various image modes and enhancements to ascertain optimal dataset characteristics, resulting in superior model performance. Through rigorous experimentation, we demonstrate that grayscale datasets, particularly without contrast enhancements, yield the highest accuracy in pattern recognition tasks. Our findings challenge existing norms in financial data analysis and propose new standards for dataset creation. The resulting publicly available dataset, a first of its kind, offers a valuable resource for future research in financial pattern analysis using deep learning. This study not only advances the field of financial analytics but also opens avenues for applying similar methodologies in other domains requiring precise pattern recognition.

Keywords Deep learning · CNN · Dataset · Pattern recognition · Stock prediction

M. El Bakai (✉) · Y. Boutyour · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team, Computer Science Laboratory (LRI), Computer Science Department, Faculty of Science of Rabat, Mohammed V University, Rabat, Morocco
e-mail: mustapha.elbakai@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_8

1 Introduction

As previously mentioned, datasets play an important and essential role in all training stages. Also it is considered the main cause of many unwanted problems, which occur during the learning process or appear as final results, and the reason is that we rely on that data without the slightest verification. This is due to several reasons, and the most important are: the absence of approved standards to be used to evaluate the credibility and effectiveness of these datasets. Also, the difficulty of creating customized datasets, because it takes time, effort, and resources.

Creating a dataset is considered one of the most important processes, because it consists of several different stages interconnected with each other, like collection, describing, labeling, etc. Some studies have shown unwanted results and negative impacts in the process of training due to dataset problems [1–3].

Our goal is to create a custom dataset to identify the existence of some chart patterns using recognition and deep learning. Patterns are recurring sequences found in the OHLC[1] chart that are used most widely in order to analyze price movements. In the research published by Bulkowski[2], he indicated the existence of relationship between patterns and price movements. Also, he observed that the repetition of some patterns is accompanied by a repetition of the same price behavior, and the main reason for using CNN models to predict price movements is that they are excellent at image treatment tasks like classification and pattern recognition, which could be lines the same as chart patterns [4].

The basic raw historical data we obtain from various trading platforms such as Binance[3], TradingView[4] and Yahoo Finance[5]…, is a time-series data, and the question that arises here is: how can we apply CNN to time-series data to solve a classification problem?

There are several ways to detect patterns in time-series using neural networks [4].

As we know, the CNN models are suitable to pattern recognition, principally because time-series data represents uses a 1-D array, and we need a solution to convert them into images. So, the GAF[6] approach can be used to convert time-series data into images [5].

Or we can simply train our model on images of chart patterns, which are not available to us, and we did not find any dataset available for the public, so we are forced to create our own dataset based on price charts, whether on trading platforms or by converting the time-series data.

During the process of creating dataset, a number of problems were addressed, which can be summarized by the following questions.

---

[1] Open, High, Low, and Close.

[2] https://thepatternsite.com/.

[3] https://www.binance.com/.

[4] https://www.tradingview.com/.

[5] https://finance.yahoo.com/.

[6] Gramian Angular Field.

1.1 What Is the Best Image Size?

Choosing the best image size is one of the most important things; this will impact the duration of creation, training time, also the performance of the model. For example, when we choose a size (width, height), we normally have to resize our current images to this new value because CNN models require inputs of identical size [6].

Therefore, every image that will occur will fall into one of the following cases:

- **Downscaling**: Is the process of downscaling or downsizing a big image with a high-resolution to get a lower resolution image and preserve its appearance [7], so larger images than the chosen size will be downsized, which will make it difficult for CNN to extract the features required for classification, since the number of pixels is reduced.
- **Upscaling**: Which is the process of enlarging a small image and increase its resolution [8]. This means that the new image will have more pixels than the original.

One of the advantages of having large-sized images in datasets is that they are rich in information and features that can be extracted and help in classification tasks. On the other hand, they consume more storage space and require large memory for pre-processing, which puts us in front of a choice between information technology efficiency and model performance.

We can also resizing it to a smaller size while maintaining an acceptable quality ratio, Whether by cropping, which may lead to the loss of features that appear in the border areas, Or by scaling, which in turn may lead to the risk of deforming features, but it is considered less risky than losing them [9].

1.2 How Much Data Do We Need?

It is necessary to have the right amount with a good quality of dataset, but the answer to the above question depends on many factors, like:

- Type and complexity of the problem,
- The complexity of the model,
- Accuracy of the data.

Some research presents a general approach to finding out the adequate size of dataset needed to have good accuracy in classification problems [10–12].

In general, the more images in the dataset, the better our model will perform. However, there are several important considerations that will affect data size and quality [13]:

- To have at least training 100 images per class, but many algorithms will require much more than this to be effective.
- Have a balanced number of images across classes.
- Split all images into train and test. A common ratio is 80/20.

1.3 Preprocessing Data

Usually we spend a lot of time on data preprocessing, which takes 50% or more of our time. For both gathering and preprocessing, quality control is an important challenge as well. Once data is collected, we have to pre-process it to make it suitable and appropriate for deep learning purposes [14]. Firstly, we check for corrupted images by deleting or replacing them.

1.4 Images Channels and Sizes

We could have images with different properties (channels, sizes, encoding formats), and we need to perform some pre-processing to make them identical.

1.5 Contrast Enhancement

It's good to convert all RGB images to grayscale if we do not need color information, or when we have images with a low contrast and we want to adjust it, we can apply:

- Histogram Equalization,
- Adaptive Histogram Equalization (AHE),
- Contrastive Limited Adaptive Histogram Equalization (CLAHE).

1.6 Normalization/Standardization

Normalization is the most important step in the preprocessing part. This refers to rescaling the pixel values to a set range often [0, 1] to help the model converge faster during training, with this equation:

$$x_{norm} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} \in [0, 1]$$

While standardization is an approach to rescaling data to be centered around the mean of 0 and a standard deviation of 1, with this equation:

$$X_{standard} = \frac{X - \mu}{\sigma}$$

1.7 Quality Control

The quality and size of the dataset play a main role in determining the performance and accuracy. In general, the more data the model has access to, the better it will perform. However, it is important to balance quantity with quality.

For example, in the financial historical data that we have, there is more than one type of price, which typically includes **Date**, **Open**, **High**, **Low**, **Volume** and **Close**, some markets provide **Adjusted Close** prices, as shown in Fig. 1.

Mostly, we rely on the closing price and not adjusted closing price to analyze market behavior and price movement. However, there is a big difference between them, as the closing price basically indicate the latest price traded on a particular trading session, which could be a minute, an hour, a day, a week…, but the adjusted closing price is a calculation adjustment made to a stock's closing price such as accounting for any corporate actions. In most cases, we obtain the same chart pattern despite the difference in the type and value of the price, as shown in Fig. 2.

But in certain cases, we can have different chart patterns for the same session and the same date, as shown in Fig. 3. Therefore, it is necessary to specify the type of price to be relied upon in the chart patterns.

1.8 Image Annotation

Sometimes we are faced with an image consisting of several chart patterns, and we need to know what does the models considers important when classifying the process. How do they make their classification decisions?

In a normal classification task, the model assigns a class to each image. However, we want to know which pixel belongs to which pattern. In this case, we need to affect a class to every pixel, this process is involved in image annotation, and this task is known as segmentation.

So we will use image annotation to know where the model is focused on patterns.

Date	Open	High	Low	Close	Adj Close	Volume
2000-01-03	36.678623	37.009064	35.215256	35.498489	7.016647	7668476
2000-01-04	35.545696	36.064957	32.854984	33.421452	6.606101	9497846
2000-01-05	33.987915	34.176739	33.563065	33.940708	6.708736	12035160
2000-01-06	33.657478	33.987915	32.571751	33.043808	6.578750	9471366
2000-01-07	33.657478	33.799095	32.760574	33.327038	6.635141	7792534

Fig. 1 AT&T Inc. stock data snapshot from January to December 2023

Fig. 2 Close versus Adjusted Close price of AT&T Inc.

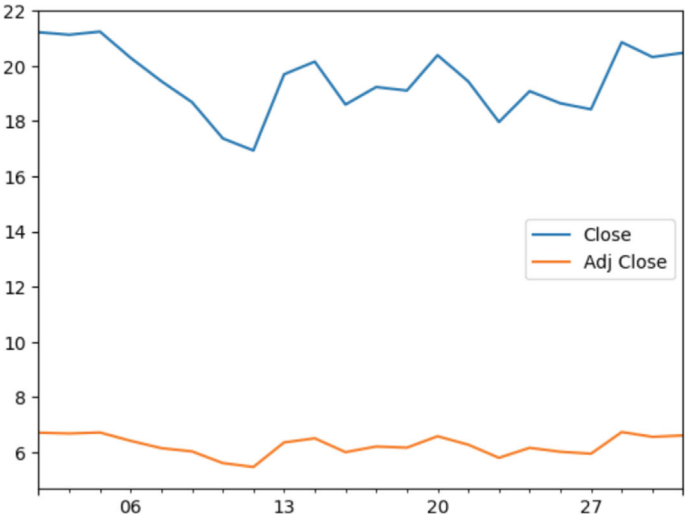


Fig. 3 Close versus Adjusted Close price of AT&T Inc. December 2008

Image annotation is the process of categorizing or labeling of various objects in an image. There are different types of techniques used for image annotation:

- **Classification annotation**: Image classification is the method of annotation that just identifies the presence of similar objects depicted in the image, sometimes known as annotation or tagging.

- **Object detection annotation**: Object detection is the combination of classification and localization to determine what objects are in an image and identify where they are by surrounding theme with a bounding box.
- **Segmentation**: This method is more advanced, because it required assigning a class to each pixel of object in image.

Image Annotation is used to reduce the search area and only focus on the important parts [15].

2 Literature Review

The existing researchs shows that the use of deep learning technology can predict the trend of Stock Market. But chart patterns recognition is still less prevalent and focuses mainly on a limited number of patterns, but some studies have worked on chart pattern recognition through many approaches and techniques.

Some studies [16–18] analyze each Candlestick position in relation to the others using indicators and investment trading strategies.

M. Velay and F. Daniel evaluates the performances of CNN and LSTM based on Gramian Angular Field for recognizing common chart patterns in stock historical data [4]. Others use the k-NN[7] algorithm to predict the market trend and check if it is bearish or bullish [19].

Chenghan Xu has another point of view from which he concluded that the GAF approach is unsuitable for pattern recognition [20].

J. Kietikul and P. Bundit indicate that using the CNN and the candlestick pattern images collected from various stocks can be used to predict the short trend for most stocks with acceptable accuracy [21].

What is notable among these studies is their reliance on candlestick charts, whether analyzing each candlestick individually or all the candlesticks forming the chart pattern. However, we used the line chart, which clearly shows the chart patterns.

In addition, we will not need the information provided by the candlestick chart; all we care about is the closing or adjusted closing price and date. Also, the chart patterns extracted from the line chart make it easy to apply image annotation processes to them, such as segmentation.

But the main problem faced is that there is no preexisting dataset with labeled chart patterns, and to overcome this obstacle, there are several solutions that vary depending on the criteria taken into consideration, which are:

- Speed,
- Quality.

For example, by building a hard-coded detector or recognizer to search in time-series data and transform the results into images [4].

[7] k-Nearest Neighbors.

Or by hand-building collected graphs directly from TradingView or any other platform by searching for chart patterns desired [22, 23].

Ha and Moon used a genetic algorithm (GA) [24] proposed by Ha and al to find all popular and profitable chart patterns [25].

Chen and Tsai used a new approach consists of two steps, calling it the GAF-CNN model. Firstly they convert time-series into images then they used it with CNN for prediction [5].

3 Methodology

Now we will describe in detail all stages of dataset creation and the tools used for this purpose.

It begins with the main problem we found, then illustrates the dataset description and the method of creation, and finally describes the model architecture.

3.1 Problem Statement

Since there is no pre-existing dataset with labeled chart patterns to use, we need to create a customized one.

We seek proven and reliable chart patterns using closing price as a reference price. In Table 1, shows the properties used.

Depending on the TradingView platform, and after adjusting the chart according to the settings shown in Table 1 above, we searched for "Double Bottom" and "Double Top" patterns manually and took a screenshot of each validated one. This process is slow, difficult, and takes a lot of time. However, the patterns are confirmed, and this is what we mentioned above when we talk about the criteria taken into consideration,

Table 1 Properties used to get images of chart patterns

Property	Name
Platform	TradingView
Stocks	AAPL, MSFT, TSLA
Chart type	Line chart
Line chart thickness	Max
Line chart color	Black
Chart background	White
Chart grid lines	None
Duration	Last 5 years
Interval	1 d
Chart patterns	Double Top, Double bottom

which are speed and quality. As a result, we took 340 images, 170 images for each chart pattern, so there were tow classes.

3.2 Dataset Description

We made sure to present our future dataset that we are in the process of creating in three parts:

- Metadata files: There are three files containing general information about:
  - Meta Class: As illustrated in Fig. 4 this file provides general information about the classes that constitute the dataset.

    It contains a unique **ClassId**,
    Path to an exemplary illustration of a typical pattern,
    ClassName to specify the class name,
    TrendId and **TrendName** to specify the trend direction, useful in result interpretation,
    Created and **Updated** to specify the date the class was included or of any modification made to it, such as adding new images, replace…,
    TrainCount and **TestCount** to specify the number of images for each class in the train and test sets.

  - Meta Train and Meta Test: As illustrated in Figs. 5 and 6, these files provide general information about images.

	Path	ClassId	ClassName	TrendId	TrendName	Created	updated	TrainCount	TestCount
0	Meta/0.jpg	0	Double top	0	bearish	02/07/2023 14:21:46	29/08/2023 16:05:57	136	34
1	Meta/1.jpg	1	Double bottom	1	bullish	02/07/2023 14:21:46	29/08/2023 16:05:57	136	34

Fig. 4 Meta file of classes

	Width	Height	Mode	Format	SizeBytes	FileSize	FileSizeUnit	ClassId	Path
0	65	105	RGB	JPEG	1673	1.63	KB	0	Train/0_0000_00081.jpg
1	97	94	RGB	JPEG	1854	1.81	KB	0	Train/0_0000_00016.jpg
2	149	88	RGB	JPEG	2387	2.33	KB	1	Train/1_0000_00126.jpg
3	94	147	RGB	JPEG	2296	2.24	KB	0	Train/0_0000_00002.jpg
4	69	122	RGB	JPEG	1652	1.61	KB	0	Train/0_0000_00094.jpg

Fig. 5 Meta file for train images

	Width	Height	Mode	Format	SizeBytes	FileSize	FileSizeUnit	ClassId	Path
0	77	140	RGB	JPEG	1347	1.32	KB	0	Test/0_00012.jpg
1	79	83	RGB	JPEG	1394	1.36	KB	0	Test/0_00033.jpg
2	56	120	RGB	JPEG	1541	1.50	KB	1	Test/1_00025.jpg
3	100	105	RGB	JPEG	1791	1.75	KB	1	Test/1_00034.jpg
4	93	119	RGB	JPEG	2431	2.37	KB	0	Test/0_00031.jpg

Fig. 6 Meta file for test images

The **width** and **height** to specify the size of the images,
The **mode** (RGB, RGBA, L…) to specify the number of colors that can be displayed,
Image **format** such as JPEG, GIF, PNG…
SizeBytes, **FileSize**, **FileSizeUnit** to specify the image size,
ClassId a unique ID to specify the class,
The **Path** where the image file is stocked,

- The raw image file was saved without any modification,
- The segmented images are used to detect and define the edges, boundaries, and outlines within an image that will be used in analysis processes for further processing, such as recognition. There are many tools to perform image segmentation; in fact, we used two tools for this purpose. At first, we used **Sefexa**[8] which is a free tool for Windows OS only, and it uses a digital pen or brush that allows us to manually annotate the different entities of an image. Then we relied on **Label Studio**[9] platform, which is an open-source data labeling tool.

We can see that all the acquired images are of different shapes and sizes, with no duplicate images, which increases the difficulty of effective classification. So in order to effectively perform the classification process, image preprocessing must be performed, which includes the conversion of all images to a standard size, then normalizing each pixel value between 0 and 1, to make the calculation process easier.

So the most important objective is to choose the best image size, because many studies have clarified that the model's performance can be proportionally affected by the resolution of the images [26] as well as by other factors like sharpening, compression, blurring, contrast…, which affect the visual information contained in the images [27].

[8] http://www.fexovi.com/sefexa.html.
[9] https://labelstud.io.

So, most resolutions used to train CNN models usually range between 64×64 and 256×256 pixels [28]. But Sabottke and al. Showed in their work that the best accuracy was achieved with lower input image resolutions [26], may be this is because lowering the number of parameters that need to be optimized reduces the risk of model over-fitting [29].

We know that the level of details perceptible in the image increases with higher resolution and is lost when down-sampling, so choosing the best image size depend on the dataset.

Now let's take a look at our dataset image sizes. Let's take a look at the sizes of the images that we have. From the Fig. 7, it becomes clear to us that we have images with varying sizes and resolutions ranging from 51×51 to 460×384 pixels, and most of them are less than 150×150 pixels.

After zooming in on the area that groups the majority of image sizes, as shown in Fig. 8, we can clearly see that most of them are grouped around either 80 or 100

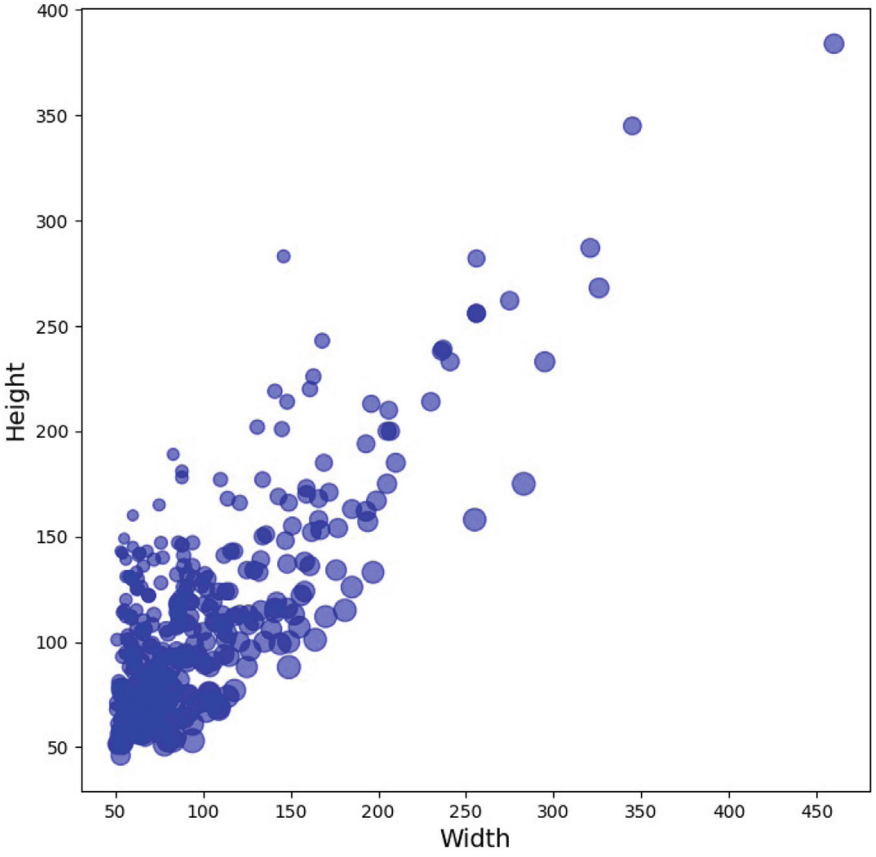


Fig. 7 Distribution of image sizes

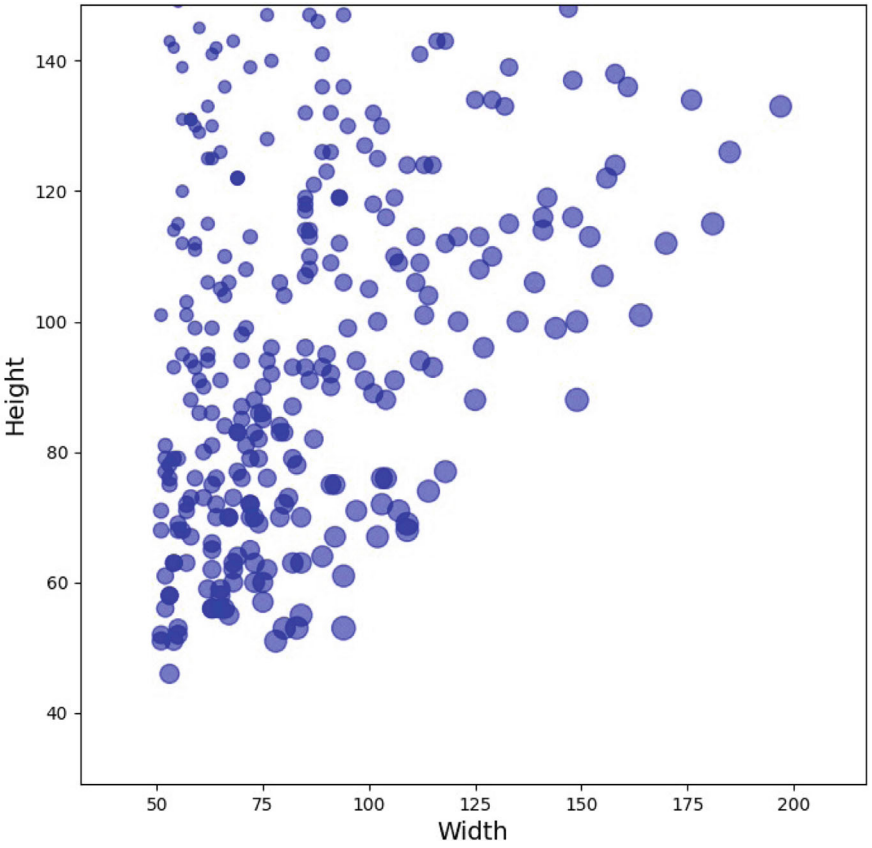


Fig. 8 Distribution of image sizes after zooming in

pixels. Also, the median value of all image sizes is 93.

$$\text{Median: } \begin{array}{l} \textit{If } n \textit{ is even then} \tilde{x} = \frac{(\frac{n}{2})^{\text{th}} \text{ obs.} + (\frac{n+1}{2})^{\text{th}} \text{ obs.}}{2} \\ \textit{If } n \textit{ is odd then} \tilde{x} = \frac{n+1}{2}^{\text{th}} \text{ obs.} \end{array}$$

So the recommended image size for this dataset is 96×96 pixels, but I suggest training our model with both RGB and grayscale mode and checking the results to confirm our choice, as well as to verify whether it is the appropriate choice and the optimal combination between resolution and the available memory?

It is possible to work on RGB or monochrome images and apply several adjustments and enhancements, such as histogram equalization as mentioned above in

Table 2 Final dataset partition

Dataset	D. Top	D. Bottom	Total	Percentage (%)
Train	136	136	272	80
Test	34	34	68	20
Total	170	170	340	100

Sect. 1.5, and the objective of this improvement is to have several datasets with different sizes and different types of enhancements to find the right dataset to use in training. Also, all models are trained using different enhancements.

In Table 2, summarizes the dataset.

3.3 Model Architecture

If we search, we will find several CNN architectures that are currently used in solving classification problems. The difference between them lies in the structure, which can be considered the essential and most important factor in improving performance. And among the most popular CNN architectures, we find the AlexNet, the High Resolution (HR), Residual Networks (ResNets), DenseNet…

Normally we had to use these models in training to benefit from their performance and their network depth…; however, they have some restrictions that will prevent us from using all image sizes. For example, the input to AlexNet is an RGB with 256×256 pixels. This means all input images must be resized to 256×256. So in this case, we cannot test a 96 × 96 dataset with grayscale enhancement.

For these reasons, the architecture of the model used in this study has been created from scatch by sequential method of keras.model and keras.layers, and its include multiple hidden layer which composed by four convolutional layers with window shape (kernel_size) of size 3×3 and the size of the filters are 32, 64, 128, 256 respectively, also we have four max-pooling layers with pool_size 2 × 2 applied after each convolutional layer, also we have a dropout which is fixed on 0.5 and tow dense layers the first is set on 1024 but the last one is set to 2.

For this network, we use a batch size of 64 and 100 epochs with the early_stopping option to stop early training when there is no more improvement in the monitored metric.

3.4 Experimental Setup

The models are implemented using Keras and TensorFlow version 2.14.0 and Python language version 3.11.6. All experiments are carried out on the jupyterLab notebook version 4.0.7 installed on Docker version 4.21.1. In Table 3, shows the different libraries and modules used in this study.

Table 3 Libraries and modules used with their versions

Libraries and modules	Version
tensorflow	2.14.0
keras	2.14.0
numpy	1.24.4
sklearn	1.3.1
skimage	0.22.0
matplotlib	3.8.0
plotly	5.17.0
pandas	2.1.1
csv	1.0
PIL	10.1.0
binance	1.0.19
yfinance	0.2.35
Python	3.11.6

The used hardware is a laptop machine with an i7 CPU, 16 GB of memory (RAM) with MS Windows-10 64 bits OS.

4 Results

As we mentioned previously, we have determined the size of dataset that we will rely on in this study, which is 96×96 pixels.

We also decided to work on both mode RGB and grayscale. For grayscale (L) mode, we applied three types of contrast enhancement, which are Histogram Equalization (HE), Adaptive Histogram Equalization (AHE) and Contrastive Limited Adaptive Histogram Equalization (CLAHE). And for the RGB we will apply only the histogram equalization (HE).

So we will have 6 datasets: L, L–HE, L–AEH, L–CLAHE, RGB and RGB–HE.

The first observation is as shown in Fig. 9 below: the larger size of the datasets, the greater their weight, which can reach up to 2 GB, and the longer the dataset creation period (Fig. 10). Also, grayscale mode is always heavier than RGB mode. Some enhancements increase weight, in particular Histogram Equalization (HE),

According to the results listed in Table 4, which shows the performance of the different datasets, we can notice the large difference in weights between the modes of RGB (72 Mo) and grayscale (24 Mo) datasets, $\frac{1}{3}$ and this reflects on the duration of training, which was completed in less time. Also, as for performance, grayscale datasets topped the results with a score of 94.71%.

A large number of studies in the existing literature ignore the effect of contrast enhancement. However, it gives us the best accuracy. Also, the top accuracy value is

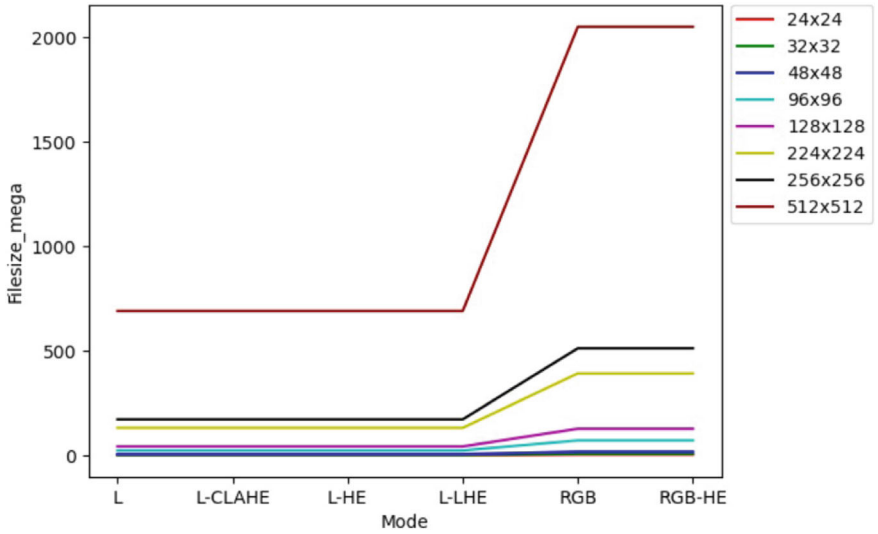


Fig. 9 The size of datasets depends on the image mode

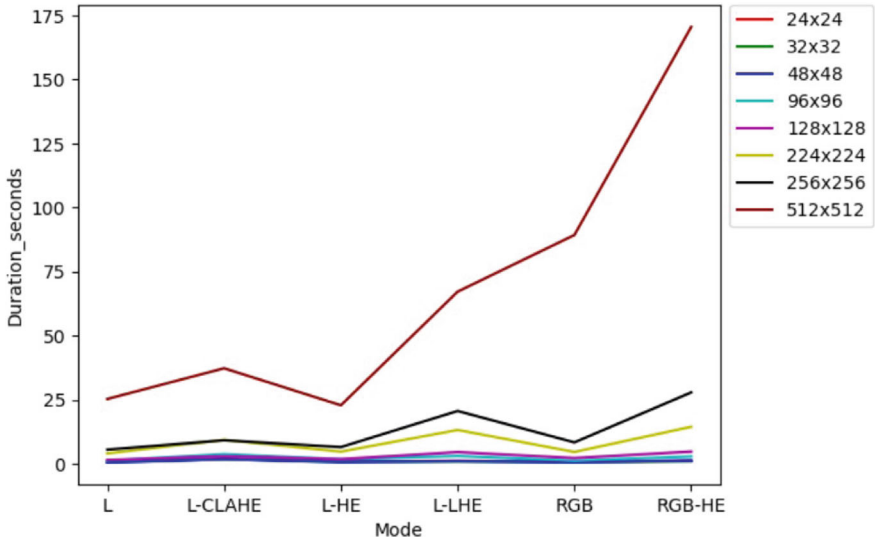


Fig. 10 Dataset creation times per mode

Table 4 The results of the different datasets

Dataset	Size (MB)	Accuracy (%)	Duration (s)
96 × 96-RGB	72.1	89.71	325.81
96 × 96-RGB-HE	72.1	90.63	284.22
96 × 96-L	24.3	94.71	173.81
96 × 96-L-HE	24.3	92.27	159.10
96 × 96-L-LHE	24.3	91.19	171.00
96 × 96-L-CLAH	24.3	91.67	119.71

observed as 94.71% using a dataset with grayscale mode without enhancement. This study highlights the importance of having several datasets with the same data but with different enhancements, ameliorations, and improvements to choose the sets that give the best performance.

5 Discussion

When we talk about charts in financial markets, the Japanese candlestick chart comes to mind, and this view is supported by several studies and research that use the candlestick charts, which unfortunately gives the impression that it is the only type of chart that exists.

The same thing for the type of price at which the chart is drawn; most of these studies depend on Close price; this information (type of price and chart) forms the basis of the graphical models on which the data is built. Can we say that this is a good choice, especially since there are several other types, whether in terms of prices or charts?

These questions and many others can only be answered through studies and comparative results. However, it should be noted that choosing Japanese candles can be a good choice in the case of technical analysis due to the data provided by the candles (Open, High, Low and Close prices), or if we are looking for a pattern composed only of a candle such as the Hanging man, Hummer…, or a sequence of candles such as Morning star, Evening Star…

In the case of other patterns such as Cup and Handle, Head and shoulders…, I think that the line chart is more indicative and easier in the search process because the information provided by the Japanese candlesticks does not matter as long as the shape is geometric.

Also, using a line chart can be compatible with creating datasets with small sizes, like 24 × 24 pixels, because an image of a chart pattern is made up only of lines that do not lose their characteristics despite the reduction in the size of the images, and this will facilitate processing even in the case of low computer resources.

Also, regarding norms and standards, I think there must be two types: general and specific. The general standards concern the description and structural design of datasets, but the other concerns the regularization linked to the activity or specialty. For example, medical data cannot be compared or processed like astronomical or financial data; each specialty has its particularities.

6 Conclusion

In this study, which aims primarily at chart pattern recognition, we have explained the different steps that preceded the creation of the datasets. We have listed the lack that exists at this stage, especially the clear absence of standards and norms that organize the creation and describing of dataset, which is considered the most important element in training.

We also created several datasets with different modes and improvements with the aim of finding the optimal combination between mode and enhancement to have the best accuracy. In this experiment, grayscale mode was the most appropriate because it gave the best results in the shortest time.

This work is not finished yet, and there is still a lot to do, including expand the size of the dataset by adding new images despite the difficulties of this step, as well as add new chart patterns. We also plan to use them to test the methods published in [30–47] for their validations.

The current dataset used in this article was released to the public on October 16, 2023, and is accessible on Kaggle platform through the link: https://www.kaggle.com/datasets/mustaphaelbakai/stock-chart-patterns (accessed January 17, 2024).

References

1. Giner-Miguelez, J., Gómez, A., Cabot, J.: DescribeML, Conference: International Conference on Model Driven Engineering Languages and Systems (MODELS). At: Montreal (Oct 2022). https://doi.org/10.1145/3550356.3559087
2. Paullada, A., Raji, I.D., Bender, E.M., Denton, E., Hanna, A.: Data and its (dis)contents: a survey of dataset development and use in machine learning research. Patterns **2**(11), 100336 (2021). https://doi.org/10.1016/j.patter.2021.100336
3. Renggli, C., Rimanic, L., Gürel, N., Karlaš, B., Wu, W., Zhang, C.: A data quality-driven view of mlops. IEEE Trans. Knowl. Data Eng. (Mar 2021)
4. Velay, M., Daniel, F.: Stock chart pattern recognition with deep learning. arXiv (Cornell University) (Aug 2018). https://doi.org/10.48550/arxiv.1808.00418. http://arxiv.org/abs/1808.00418
5. Chen, J., Tsai, Y.: Encoding candlesticks as images for pattern classification using convolutional neural networks. Financ. Innov. **6**(1) (Jun 2020). https://doi.org/10.1186/s40854-020-00187-0
6. Hashemi, M.: Web page classification: a survey of perspectives, gaps, and future directions. Multimed. Tools Appl. **79**(17–18), 11921–11945 (2020). https://doi.org/10.1007/s11042-019-08373-8

7. Sun, W., Chen, Z.: Learned image downscaling for upscaling using content adaptive resampler. IEEE Trans. Image Process. **29**, 4027–4040 (2020). https://doi.org/10.1109/tip.2020.2970248
8. Dong, C., Loy, C.C., He, K., Tang, X.: Image Super Resolution using deep convolutional networks. arXiv (Cornell University) (Dec 2014). https://doi.org/10.48550/arxiv.1501.00092
9. Hashemi, M.: Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation. J. Big Data **6**(1) (Nov 2019). https://doi.org/10.1186/s40537-019-0263-7
10. Chen, H.: Maybe only 0.5% data is needed: a preliminary exploration of low training data instruction tuning (May 2023). https://arxiv.org/abs/2305.09246v1
11. Shahinfar, S., Meek, P.D., Falzon, G.: How many images do I need? Understanding how sample size per class affects deep learning model performance metrics for balanced designs in autonomous wildlife monitoring. Ecol. Inform. **57**, 101085 (2020). https://doi.org/10.1016/j.ecoinf.2020.101085
12. Cho, J.: How much data is needed to train a medical image deep learning system to achieve necessary high accuracy? (Nov 2015). https://arxiv.org/abs/1511.06348
13. 7 Image datasets for classification & how to build your own (May 2023). https://datagen.tech/guides/image-datasets/image-dataset-for-classification/
14. Roh, Y., Heo, G., Whang, S.E.: A survey on data collection for machine learning: a big data—AI integration perspective. IEEE Trans. Knowl. Data Eng. **33**(4), 1328–1347 (2021). https://doi.org/10.1109/tkde.2019.2946162
15. Boesch, G.: Image annotation: best software tools and solutions in 2024 (Dec 2023). https://viso.ai/computer-vision/image-annotation/
16. Lin, Y., Liu, S., Yang, H., Wu, H., Jiang, B.: Improving stock trading decisions based on pattern recognition using machine learning technology. PLoS One **16**(8), e0255558 (2021). https://doi.org/10.1371/journal.pone.0255558
17. Thammakesorn, S., Sornil, O.: Generating trading strategies based on candlestick chart pattern characteristics. In: Journal of Physics: Conference Series, vol. 1195, p. 012008 (2019). https://doi.org/10.1088/1742-6596/1195/1/012008
18. Tripathi, A., Mathure, J., Deotarse, S., Gadhikar, D.R.L.: Linear regression approach for stock chart pattern recognition (Jan 2023). https://ieeexplore.ieee.org/document/10146731
19. Subha, M., Nambi, S.: Classification of stock index movement using k-nearest neighbours (k-nn) algorithm. WSEAS Trans. Inf. Sci. Appl. **9**, 261–270 (2012)
20. Liu, L., Si, Y.-W.: 1D convolutional neural networks for chart pattern classification in financial time series. J. Supercomput. **78**(12), 14191–14214 (2022). https://doi.org/10.1007/s11227-022-04431-5
21. Kietikul Jearanaitanakij, B.P.: Predicting short trend of stocks by using convolutional neural network and candlestick patterns. In: IEEE Conference Publication. IEEE Xplore (Oct 2019). https://ieeexplore.ieee.org/document/8912115
22. Kaya, C.B., Yılmaz, A., Uzun, G.N., Kilimci, Z.H.: Stock pattern classification from charts using deep learning algorithms. Acad. Perspect. Procedia 3(1), 445–454 (2020). https://doi.org/10.33793/acperpro.03.01.89 https://doi.org/10.33793/acperpro.03.01.89
23. Chen, J.-H., Tsai, Y.-C.: Dynamic deep convolutional candlestick learner (Jan 2022). https://arxiv.org/abs/2201.08669
24. Ha, M.H., Moon, B.-R.: The evolution of neural network-based chart patterns: a preliminary study. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO '17. Association for Computing Machinery, New York, NY, USA (2017), pp. 1113–1120. https://doi.org/10.1145/3071178.3071192
25. Ha, M.H., Lee, S., Moon, B.-R.: A genetic algorithm for rule-based chart pattern search in stock market prices. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, GECCO '16. Association for Computing Machinery, New York, NY, USA (2016), pp. 909–916. https://doi.org/10.1145/2908812.2908828
26. Sabottke, C., Spieler, B.: The effect of image resolution on deep learning in radiography. Radiology **2**(1), e190015 (2020). https://doi.org/10.1148/ryai.2019190015

27. Sheikh, H., Bovik, A.C.: Image information and visual quality. IEEE Trans. Image Process. **15**(2), 430–444 (2006). https://doi.org/10.1109/tip.2005.859378
28. Thambawita, V., Strümke, I., Hicks, S.A., Halvorsen, P., Parasa, S., Riegler, M.: Impact of image resolution on deep learning performance in endoscopy image classification: an experimental study using a large dataset of endoscopic images. Diagnostics **11**(12), 2183 (2021). https://doi.org/10.3390/diagnostics11122183
29. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. IEEE Trans. Neural Netw. **5**(4), 537–550 (1994). https://doi.org/10.1109/72.298224
30. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular ad-hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
31. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv:1307.5910
32. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
33. Idrissi, A., Li, C.: Modeling and optimization of the capacity allocation problem with constraints. In: RIVF, pp. 107–116 (2006)
34. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
35. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
36. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
37. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
38. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Arch. **9**(2–3), 136–148 (2020)
39. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless ad hoc networks using the skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of Things and Cloud Computing (2016)
40. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
41. Boutyour, Y., Idrissi, A.: Deep reinforcement learning in financial markets context: review and open challenges. In: Modern Artificial Intelligence and Data Science. Springer Nature Switzerland, pp. 49–66 (2023). https://doi.org/10.1007/978-3-031-33309-5_5
42. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol., 5567–5584 (2023)
43. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
44. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
45. Handri, K.E., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
46. Handri, K.E., Idrissi, A.: Comparative study of top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
47. Elhandri, K., Idrissi, A.: Parallelization of top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2021)

Bridging the Gap Between Ontology Engineering and Software Engineering

Ouassila Labbani Narsis and Christophe Nicolle

Abstract Ontology engineering requires specific skills and expertise, making its adoption by software developers a challenging task. The complex nature of ontological modeling and the use of specialized languages and tools represent significant obstacles for those unfamiliar with this domain. Consequently, bridging the gap between ontology engineering and software engineering requires innovative approaches that simplify the integration of ontological concepts into software systems, enabling developers to harness the benefits of ontological reasoning and semantic knowledge representation. This paper introduces a new approach that automates the transformation of ontologies into UML/OCL models, commonly used by software developers. The integration of OCL constraints enables the precise expression of axiomatic relationships and logic required to simulate the ontological reasoning process. This approach allows the translation of ontologies into a developer-friendly format while preserving the semantics and reasoning capabilities of ontological models. The transformation process is implemented in an application named ReCoRe, which is used to test and validate the approach. It represents a significant step towards enhancing the synergy between ontology and software engineering, making the adoption of ontologies in practical software development more accessible and beneficial to a wider range of developers.

1 Introduction and Motivation

Knowledge engineering (KE) is an active area of artificial intelligence research for the design and implementation of knowledge-based systems. It is defined by Feigenbaum and McCorduck in 1984 as follows: "*Knowledge engineering is an engineering discipline that involves integrating knowledge into computer systems in order to solve complex problems normally requiring a high level of human expertise*" [1]. Knowl-

O. Labbani Narsis (✉) · C. Nicolle
CIAD UMR 7533, Université de Bourgogne, UB, Dijon 21000, France
e-mail: ouassila.narsis@u-bourgogne.fr

C. Nicolle
e-mail: cnicolle@u-bourgogne.fr

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_9

edge engineering aims to propose concepts, methods, and techniques that allow the development of computer systems based on mechanisms that manipulate knowledge and its semantics. From this field emerged the concept of **ontology engineering** (OE), which is the activity aimed at constructing and operating conceptual structures of knowledge in a specific domain, in the form of a model of concepts and their relationships [2, 3].

Interest in ontology engineering has expanded into industrial settings for collecting, sharing, and formalizing knowledge from various sources. In this context, building an ontology requires using specific approaches and methodologies to transform raw, unstructured knowledge, often provided by domain experts, into a well-organized, structured, and formal knowledge representation. This formalization not only enhances knowledge management but also empowers decision-making processes within these industrial environments.

Several ontology development methodologies are available. They primarily focus on the process and methodological aspects of ontology engineering to construct ontologies from scratch at the knowledge level [4]. However, building an ontology remains a challenging process [5]. Ontologies are typically created by highly specialized ontology engineers, and there is no tool support for the methodology [6]. They are mainly based on approaches, criteria, and development guides related to the expertise of individuals in a specific domain, and are typically influenced by the intended application. Furthermore, ontology descriptions often support multiple modifications throughout their life cycle due to evolution or alignment processes, which could introduce inconsistencies into their descriptions [7]. Novice users, who lack expert knowledge, can easily become confused by the usage of ambiguous terminology and synonyms. Therefore, understanding and effectively integrating an ontology can become a complex task that requires additional skills. This complexity represents a significant barrier to the industrial adoption of semantic technologies [8].

In industry, engineers are often familiar with software technologies and lack expertise in ontology development. It is then important to propose new approaches to bridge the gap between these two domains, enhancing collaboration and fostering the development of innovative solutions based on knowledge representation.

In **software engineering** (SE), object-oriented modeling (OOM) possesses several interesting characteristics for knowledge representation and is well-suited to intuitive real-world modeling [9]. Object-oriented techniques have been widely used to address knowledge acquisition and representation issues [10]. In this field, the Unified Modeling Language (UML) is used for system modeling and visualization [11]. UML provides diagrams that are easy to understand for users and is widely adopted in the industry.

Several research works adapt UML diagrams, particularly class diagrams, for ontology development [12]. Despite their similarities, ontology and object-oriented modeling differ in several ways. The most significant difference is that the ontology is theoretically founded on description logic, allowing for automated reasoning, which is not the case for object-oriented modeling. To address this limitation, this paper proposes adding logical reasoning capability to object-oriented modeling using Object Constraint Language (OCL) [14]. We present a methodology along with rules for

converting ontology axioms into a UML/OCL model. The results of OCL constraint verification will then be used to generate new knowledge and verify the ontology's consistency.

2 Ontology Engineering and Object-Oriented Modeling

Knowledge acquisition and representation are primary research areas in ontology engineering and software development, particularly concerning the utilization of ontology languages and UML models. The similarity between these two modeling paradigms has been extensively studied in several research projects that combine UML modeling and ontologies [12]. Their objectives include proposing a graphical visualization of the ontology using UML diagrams [15–21], unifying knowledge representation models [22–24], or validating existing UML diagrams [25, 27–29]. However, no studies have addressed the perspective of using object-oriented modeling for deductive reasoning. Such reasoning, available in ontologies, can be used to generate new knowledge or verify the consistency of digitized knowledge. In [30], the authors discuss the potential of the OCL language combined with UML modeling for ontological reasoning. They claim that UML design tools can be extended with logical properties closer to description logic using OCL [31]. An inverse approach is presented in [32] with a transformation of OCL invariants into OWL 2 axioms to verify the consistency of UML models. Another approach to analyzing the consistency of UML state diagrams with OCL invariants using ontological reasoning is presented in [33]. Other research work focuses on the correspondence between query languages and OCL [34–37], and others mention OCL to represent some constraints to facilitate the correspondence between UML and ontologies [20, 38].

An ontology is often formally expressed using description logic [39], which provides a structured means to represent terminological knowledge within an application domain. Description logic can be translated into first-order predicate logic and consists of a set of axioms defining modeling constraints for the ontology's components [40]. The OCL language is also based on first-order predicate logic but offers a simplified notation similar to the programming languages [41–43]. This language is widely used to express precise, unambiguous constraints in the context of object-oriented systems, mainly in UML models. It can be used to take into account, in the form of properties, the requirements of the applications to be developed that cannot be expressed graphically using UML diagrams, thus providing an essential complement to the graphical representation of models.

Since the OCL language can be used to specify the logical properties of systems, we have explored the potential of using this language to express the logical properties of ontology axioms. This allows the development of a methodology for implementing a reasoning process based on the verification results of OCL constraints.

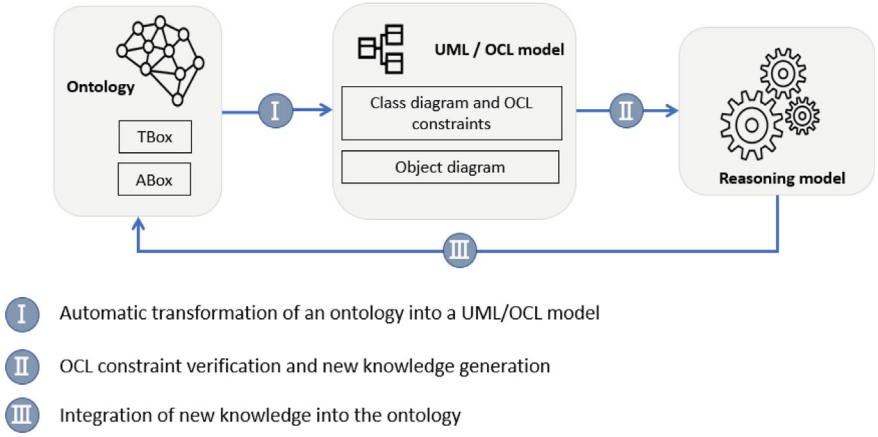


Fig. 1 Ontological modeling and reasoning process using UML/OCL modeling

3 Ontology Modeling and Reasoning Using UML/OCL Modeling

One of the main objectives of ontology engineering is to create intelligent systems capable of reasoning to extract new knowledge. In the literature, various methods and reasoning tools are presented [44]. These approaches are typically based on inference rules, commonly specified using description logic, to perform reasoning processes on classes, object properties, and assertions.

To promote the adoption of ontology engineering by software developers and exploit the potential of reasoning systems, we propose endowing UML modeling with a reasoning system based on OCL constraint verification. The goal of our approach is to simplify ontology comprehension by automatically converting it into a UML/OCL model while preserving the semantics of ontology axioms. This offers the advantage of creating a more accessible reasoning system based on UML modeling and constraint verification, which can be easily integrated by software developers. With this approach, users can also adapt and specify the model by adding specific constraints directly in the OCL language. A global view of our approach is given in Fig. 1.

There are three main steps. Step **I** aims to automatically generate the UML model and the OCL constraints related to the ontology (TBox and ABox levels). To do that, we have defined a set of transformation rules to create the UML/OCL model from the ontology description. OWL 2 defines several types of axioms to represent ontology knowledge.[1] In our study, we focused on transforming *Declaration*, *ClassAxiom*, *ObjectPropertyAxiom*, and *Assertion* axioms.

The transformation process involves two stages. The first one consists of generating the UML model and OCL constraints related to the ontological entities and

[1] https://www.w3.org/TR/owl2-syntax/#Axioms.

Table 1 Transformation of the `DisjointClasses` axiom

DisjointClasses	
OWL 2 axiom	UML/OCL model

`DisjointClasses(`C_1 ... C_n`)`
 	
All the classes C_i, $1 \leq i \leq n$, are pairwise disjoint; that is, no individual can be at the same time an instance of both classes C_i and C_j, $i \neq j$.
	Add an OCL constraint for each pair of classes C_i and C_j, $i \neq j$, `of the following form:`
	`context Thing inv:`
	`not(oclIsKindOf(`C_i`) and oclIsKindOf(`C_j`))`
	
	This constraint ensures that an individual cannot be of both types C_i and C_j simultaneously.

their relationships. Table 1 shows an example of generated OCL constraints when transforming the `DisjointClasses` axiom.[2]

The second stage aims to describe the logical relationships between entities by generating OCL constraints that express the semantics of each axiom with other ontology axioms. For example, if the ontology defines `DisjointClasses(`C_i C_j`)` and `SubClassOf(`C_k C_i`)` axioms, it is necessary to ensure that the semantics of these two axioms guarantee that C_k is disjoint from C_j and all classes equivalent to C_j. In this case, the following OCL constraints are automatically generated and added to the UML model:
`context Thing inv:`
`not(oclIsKindOf(`C_k`)and oclIsKindOf(`C_j`))`,
and
`context Thing inv:`
`not(oclIsKindOf(`C_k `)and oclIsKindOf(`C_m`))`, for each C_m equivalent to C_j.

After generating the UML/OCL model, step **II** in the global process enables verification of the OCL constraint and analysis of the verification results to detect new logical relationships and generate associated knowledge. In UML modeling, when an OCL constraint is not verified, it may be due to either an error in the model's design or a lack of knowledge that can be automatically generated and integrated into the ontology. In the previous example, if the OCL constraints related to the definition of a `SubclassOf` axiom with a `DisjointClasses` axiom are not verified, the application generates the following new knowledge:
`DisjointClasses(`C_k C_j`)` and `DisjointClasses(`C_k C_m`)`, for each C_m equivalent to C_j.

[2] For simplification reasons, and due to the paper's page limit, all transformation rules are not detailed.

Finally, Step **III** allows the integration of new knowledge generated, after user validation, into the ontology. This process is iterative and continues until there is no new knowledge to generate, and all OCL constraints have been successfully verified.

4 Implementation and Experimentation

To validate and test our approach, we implemented the proposed transformation rules in an application named ReCoRe (**Re**asoning by **Co**nstraint **Re**solution), which automates all the steps of the process described in Fig. 1.

Figure 2 illustrates the global architecture of the ReCoRe application.

It consists of a *backend* component responsible for implementing the transformation rules of the ontology's axioms into a UML/OCL model and generating new knowledge based on the result of constraints verification. The backend component also supports the integration of the USE[3] application for constraint verification and provides the results as an application programming interface (API) using the open-source Swagger[4] framework. The *frontend* component is the visual part of the ReCoRe application, offering an interface that enables users to interact with the application. Users can load an ontology, convert it into a UML/OCL model, display and add new knowledge, or visualize UML diagrams using the open-source PlantUML[5] application. To operate, the frontend component calls the different functions proposed by the backend API.

To illustrate our approach, we propose to study a simple example of an ontology presented in Fig. 3. In this ontology, we have defined the following axioms:

```
EquivalentClasses(Human Person)
DisjointClasses(Person Animal)
DisjointUnion(Child Boy Girl)
SubObjectPropertyOf (hasDog, hasPet)
```

The first step of the ReCoRe application involves automatically generating the UML model and the associated OCL constraints. These constraints are of two types, as described in Sect. 3. The first type concerns all OCL constraints related to the semantics of ontology axioms that describe classes, object properties, and assertions. In the studied example, the `SubClassOf(Child Human)` axiom, for instance, which means that class `Child` is a subclass of class `Human`, is automatically transformed and represented in the UML model by a generalization link between classes `Child` and `Human`, and the following OCL constraint:

```
context Child inv:
self.oclIsKindOf(Human)
```

[3] https://sourceforge.net/projects/useocl.

[4] https://swagger.io.

[5] https://plantuml.com.

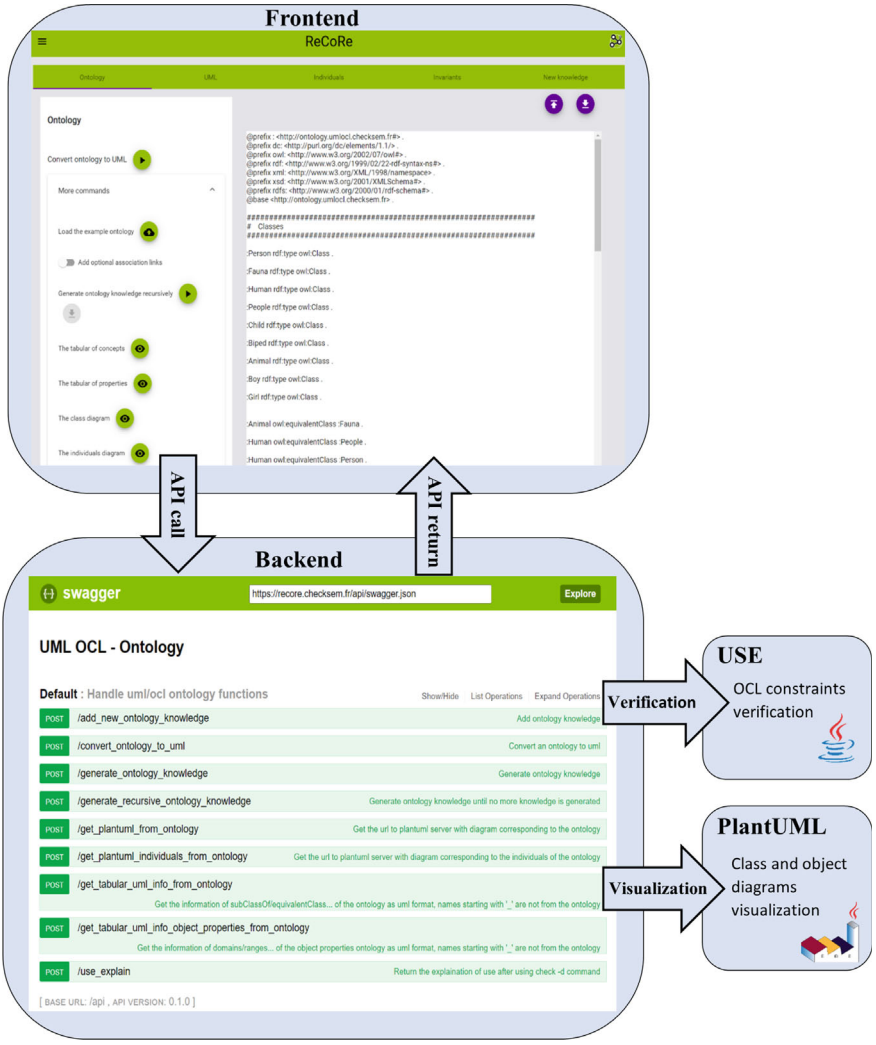


Fig. 2 Global architecture of the ReCoRe application

The second type of OCL constraint concerns the semantics of each axiom concerning the other axioms defined in the ontology. In the studied example, the semantics of the `SubClassOf(Child Human)` axiom combined with the `EquivalentClasses(Human Person)` axiom, allow for the automatic generation of the following OCL constraint, which means that all individuals of type `Child` must also be of type `Person`:

```
context Child inv:
self.oclIsKindOf(Person)
```

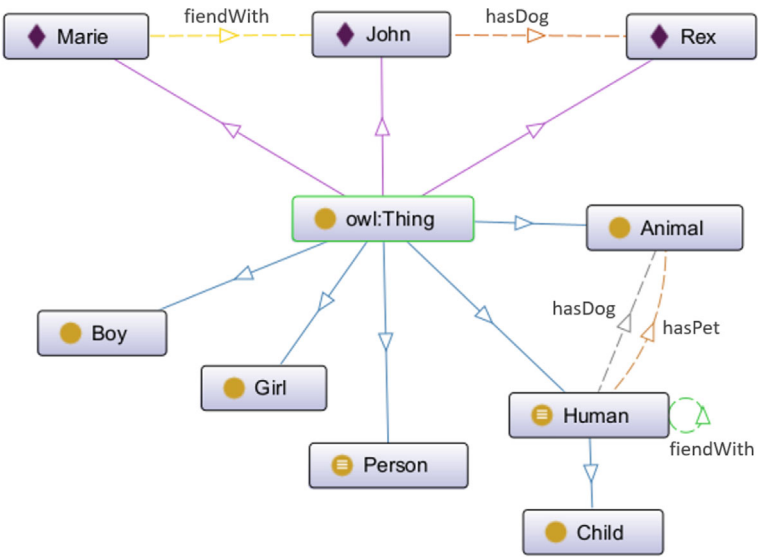


Fig. 3 Simple example of an ontology

The generated UML model can be visualized using the PlantUML component integrated with the ReCoRe application. This enables the display of the UML class diagram representing the taxonomy of classes and the object diagram representing the set of individuals, as shown in Fig. 4. It is also possible to download the model generated by our application in the *.use* format and open it using the USE application to visualize the UML model and all associated OCL constraints.

After generating the UML/OCL model, the second step in the ReCoRe application consists of automatically verifying the generated OCL constraints and, based on the verification results, automatically generating associated knowledge. In this approach, verifying constraints related to axiom semantics endows the UML model with ontological reasoning capabilities without having to use ontological reasoners. Figure 5 represents a subset of the knowledge generated by the ReCoRe application for the studied example. Each generated knowledge is explained, and the user can select which knowledge to integrate into the ontology based on their specific requirements. This explainability part is essential to assist users in better integrating the reasoning results. It is presented in a simple natural language that is easily understandable even for non-experts in the field of ontological reasoning. We have compared all the knowledge generated by the ReCoRe application with that provided by the Pellet[6] reasoner within the Protégé tool.[7] The results showed a strong correspondence, which demonstrates the reliability and feasibility of our approach,

[6] https://www.w3.org/2001/sw/wiki/Pellet.

[7] https://protege.stanford.edu/.

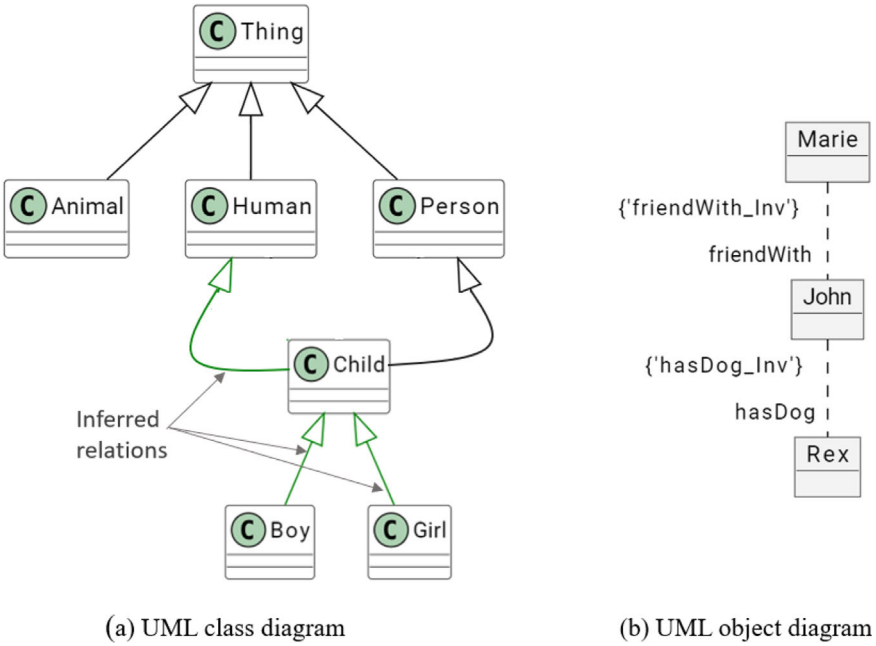


Fig. 4 UML model automatically generated from the ontology example of Fig. 3

and its practical utility for software developers in adopting ontology engineering and reasoning.

5 Conclusion and Future Work

Knowledge engineering methods and tools, in particular ontological modeling, are not widely used and represent a new challenge for software developers. To facilitate the adoption of these new technologies, in this paper, we have proposed an ontological modeling and reasoning approach based on the object paradigm, widely used in software engineering. Our approach consists of transforming the ontology into a UML model with associated OCL constraints according to the semantics of the ontology's axioms. This allows graphical UML models to be endowed with the logical properties needed in the reasoning process.

We have implemented the transformation process and shown that the result of OCL constraint verification can be used to automatically generate new knowledge from an ontology. Initial results show the feasibility of our approach and the relevance of UML/OCL modeling to ontological reasoning. The proposed approach also enables domain experts to understand the result of the reasoning process and to validate and select knowledge to be integrated into the ontology according to the usage

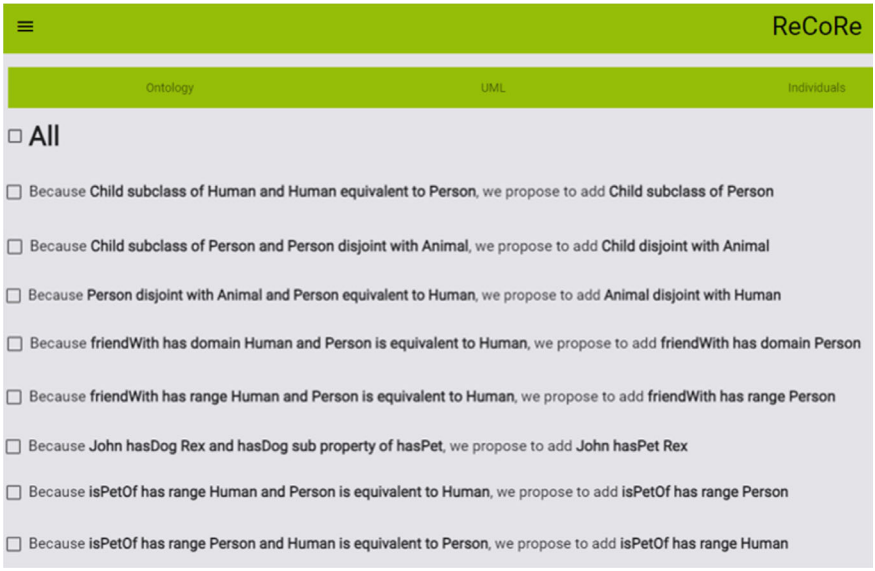


Fig. 5 Example of generated knowledge using the ReCoRe application

requirements and context. It also enables the detection of inconsistencies within the ontology and can serve as a support for ontology construction for non-domain experts.

Future work will finalize the transformation rules for all ontology axioms, and compare them with other ontology reasoning models. A generic constraint expression model will also be proposed to help users in the specification of new constraints for more effective management and specialization of existing ontologies.

Acknowledgements The authors thank Nicolas Gros for his contribution to the implementation of the ReCoRe application.

References

1. Feigenbaum, E., McCorduck, P.: The Fifth Generation. Pan Books London (1984)
2. Corcho, O., Gómez-Pérez, A., Fernàndez-López, M.: Ontological Engineering. With Examples From the Areas of Knowledge Management, E-Commerce and the Semantic Web (Advanced Information And Knowledge Processing) (2004)
3. Mizoguchi, R.: Tutorial on ontological engineering Part 2: ontology development, tools and languages. New Gener. Comput. **22**, 61–96 (2004)
4. Fernàndez-López, M., Gómez-Pérez, A., Juristo, N. : Methontology: From Ontological Art Towards Ontological Engineering. American Asociation for Artificial Intelligence (1997)
5. Stadnicki, A., Pietroń, F., Burek, P.: Towards a modern ontology development environment. Procedia Comput. Sci. **176**, 753–762 (2020)

6. Dahlem, N., Hahn, A. : User-friendly ontology creation methodologies-a survey. In: AMCIS 2009 Proceedings, p. 117 (2009)
7. Behkamal, B., Naghibzadeh, M.: Inconsistency repair to improve the alignment results of ontology matchers. Int. J. Inf. Commun. Technol. Res. **9**, 17–23 (2017)
8. Lupp, D., Hodkiewicz, M., Skjæveland, M.: Template libraries for industrial asset maintenance: a methodology for scalable and maintainable ontologies. In: CEUR Workshop Proceedings, vol. 2757, pp. 49–64 (2020)
9. Engels, G., Groenewegen, L.: Object-oriented modeling: a roadmap. In: Proceedings of the Conference on the Future of Software Engineering, pp. 103–116 (2000)
10. Dai, H., Hughes, J., Bell, D.: Knowledge representation and problem-solving using object-oriented paradigm. In: Proceedings Of TENCON'93. IEEE Region 10 International Conference on Computers, Communications and Automation, vol. 1, pp. 275–279 (1993)
11. OMG: Unified Modeling Language (OMG UML): Version 2.5.1, 2017 Object Management Group, Inc. (2017). https://www.omg.org/spec/UML
12. Mkhinini, M., Labbani-Narsis, O., Nicolle, C.: Combining UML and ontology: an exploratory survey. Comput. Sci. Rev. **35**, 100–223 (2020)
13. Siricharoen, W.: Ontologies and object models in object oriented software engineering. IAENG Int. J. Comput. Sci. **33**, 19–24 (2007)
14. OMG: Object constraint language. Version 2.4 (2014). https://www.omg.org/spec/OCL/About-OCL
15. Cranefield, S., Purvis, M.: UML as an Ontology Modelling Language. University of Otago (1999)
16. Liepinš, R., Grasmanis, M., Bojars, U.: OWLGrEd ontology visualizer. In: Proceedings of the 2014 International Conference on Developers, vol. 1268, pp 37–42 (2014)
17. Gašević, D., Djurić, D., Devedžić, V.: MDA-based automatic OWL ontology development. Int. J. Softw. Tools Technol. Transfer. **9**, 103–117 (2007)
18. Chung, S., Tai, W., OSullivan, D., Boran, A. : A semantic mapping representation and generation tool using UML for system engineers. In: IEEE International Conference on Semantic Computing, pp. 235–241 (2014)
19. Almeida Ferreira, D., Silva, A.: UML to OWL mapping overview an analysis of the translation process and supporting tools. In: 7th Conference of Portuguese Association of Information Systems (2013)
20. Baclawski, K., Kokar, M., Kogut, P., Hart, L., Smith, J., Holmes III, W., Letkowski, J., Aronson, M.: Extending UML to support ontology engineering for the semantic web. International Conference on the Unified Modeling Language, pp. 342–360 (2001)
21. Brockmans, S., Volz, R., Eberhart, A., Löffler, P.: Visual modeling of OWL DL ontologies using UML. In: The Semantic Web-ISWC 2004: Third International Semantic Web Conference, Hiroshima, Japan, November 7–11, 2004. Proceedings 3, pp. 198–213 (2004)
22. Elasri, H., Elabbassi, E., Abderrahim, S., Fahad, M.: Semantic integration of UML class diagram with semantic validation on segments of mappings (2018). ArXiv:1801.04482
23. Robles, K., Fraga, A., Morato, J., Llorens, J.: Towards an ontology-based retrieval of UML Class Diagrams. Inf. Softw. Technol. **54**, 72–86 (2012)
24. Keet, C., Fillottrani, P.: An ontology-driven unifying metamodel of UML Class Diagrams, EER, and ORM2. Data Knowl. Eng. **98**, 30–53 (2015)
25. He, H., Wang, Z., Dong, Q., Zhang, W., Zhu, W.: Ontology-based semantic verification for UML behavioral models. Int. J. Softw. Eng. Knowl. Eng. **23**, 117–145 (2013)
26. Khan, A., Musavi, S., Rehman, A., Shaikh, A.: Ontology-based finite satisfiability of UML class model. IEEE Access **6**, 3040–3050 (2018)
27. Khan, A., Porres, I.: Consistency of UML class, object and statechart diagrams using ontology reasoners. J. Vis. Lang. Comput. **26**, 42–65 (2015)
28. Elsayed, E., El-Sharawy, E.: Detecting design level anti-patterns; structure and semantics in UML Class Diagrams. J. Comput. **13**, 638–654 (2018)
29. Sadowska, M., Huzar, Z.: Semantic validation of UML class diagrams with the use of domain ontologies expressed in OWL 2. In: Software Engineering: Challenges and Solutions: Results

of the XVIII KKIO 2016 Software Engineering Conference 2016 Held At September 15–17 2016 in Wroclaw, Poland, pp. 47–59 (2017)
30. Cranefield, S., Purvis, M.: UML as an ontology modelling language. Intelligent Information Integration (1999)
31. Cranefield, S., Haustein, S., Purvis, M.: UML-based ontology modelling for software agents. Citeseer (2001)
32. Fu, C., Yang, D., Zhang, X., Hu, H.: An approach to translating OCL invariants into OWL 2 DL axioms for checking inconsistency. Autom. Softw. Eng. **24**, 295–339 (2017)
33. Khan, A., Porres, I.: Consistency of UML class, object and statechart diagrams using ontology reasoners. J. Vis. Lang. Comput. **26**, 42–65 (2015)
34. Parreiras, F., Staab, S.: Using ontologies with UML class-based modeling: the TwoUse approach. Data Knowl. Eng. **69**, 1194–1207 (2010)
35. Hafeez, A., Musavi, S., Rehman, A.: Ontology-based verification of UML class/OCL model. Mehran Univ. Res. J. Eng. Technol. **37**, 521–534 (2018)
36. Milanović, M., Gašević, D., Giurca, A., Wagner, G., Devedžić, V.: On interchanging between owl/swrl and uml/ocl. In: Proceedings of 6th Workshop On OCL For (Meta-) Models in Multiple Application Domains (OCLApps) at the 9th ACM/IEEE International Conference On Model Driven Engineering Languages And Systems (MoDELS), Genoa, Italy, pp. 81–95 (2006)
37. Timm, J., Gannod, G. : Specifying semantic web service compositions using UML and OCL. IEEE International Conference On Web Services (ICWS 2007). pp. 521–528 (2007)
38. Wang, X., Chan, C.: Ontology modeling using UML. OOIS **2001**, 59–68 (2001)
39. Baader, F., Horrocks, I., Sattler, U.: Description logics. Found. Artif. Intell. **3**, 135–179 (2008)
40. Hustadt, U., Schmidt, R., Georgieva, L.: A survey of decidable first-order fragments and description logics. J. Relat. Methods Comput. Sci. **1**, 3 (2004)
41. Franconi, E., Mosca, A., Oriol, X., Rull, G., Teniente, E.: Logic foundations of the OCL modelling language. In: Logics In Artificial Intelligence: 14th European Conference, JELIA 2014, Funchal, Madeira, Portugal, September 24–26, 2014. Proceedings 14, pp. 657–664 (2014)
42. Kanso, B., Taha, S.: Specification of temporal properties with OCL. Sci. Comput. Program. **96**, 527–551 (2014)
43. Beckert, B., Keller, U., Schmitt, P.: Translating the Object Constraint Language into first-order predicate logic. In: Proceedings Of The Second Verification Workshop: VERIFY, vol. 2, pp. 02–07 (2002)
44. Abburu, S.: A survey on ontology reasoners and comparison. Int. J. Comput. Appl. **57** (2012)

Artificial Intelligence, Computer Vision and NLP

A Re-assessment of Code2Vec

Oumaima Bel Moudden, Rym Guibadj, Denis Robilliard, Cyril Fonlupt, Abdeslam Kadrani, and Rachid Benmansour

Abstract Code2Vec has emerged as a powerful tool for analyzing source code by leveraging distributed representations. Code2Vec has demonstrated substantial capabilities in capturing semantic information from source code; however, its sensitivity to variable names has been identified as a significant limitation. This sensitivity raises concerns about the robustness of the model's performance in different code bases with varying naming conventions. In response to this limitation, our study focuses on evaluating the impact of a variable name anonymisation technique. The anonymization process was guided by the fundamental concept that a program's semantics remain consistent despite changes in variable names. This idea underscores the potential for exploring Code2Vec's learning performance with anonymous variables. As operations within a program constitute its semantics, and certain paths encapsulate these operations, it prompts a natural inquiry into whether Code2Vec's learning process can be enhanced by prioritizing operations over variable names. By anonymizing variable names, we aim to enhance Code2Vec's generalization performance, especially when confronted with non-human-generated source code, such as that produced by genetic programming.

O. Bel Moudden (✉) · R. Guibadj · D. Robilliard · C. Fonlupt
Université Littoral Cote d'Opale, UR 4491, LISIC, Laboratoire d'Informatique Signal et Image de la Cote d'Opale, Calais, France
e-mail: oumaima.bel-moudden@univ-littoral.fr

R. Guibadj
e-mail: rym.guibadj@univ-littoral.fr

D. Robilliard
e-mail: denis.robilliard@univ-littoral.fr

C. Fonlupt
e-mail: cyril.fonlupt@univ-littoral.fr

O. Bel Moudden · A. Kadrani · R. Benmansour
SI2M Laboratory, INSEA, University of Mohammed-V, Rabat, Morocco
e-mail: akadrani@insea.ac.ma

R. Benmansour
e-mail: r.benmansour@insea.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_10

Keywords Code embedding · Code2Vec · Code representation · Source code analysis

1 Introduction

In the context of natural language processing, word embeddings [1] have been widely used to represent words as vectors in continuous space, capturing semantic relationships between words. Similarly, in the domain of source code [2], code embedding techniques aim to represent code snippets, functions, or entire programs as vectors. Code embedding refers to the process of representing source code in a continuous vector space, usually of lower dimensionality, using embedding techniques [3]. There are several software engineering applications of code embedding: detecting duplicates or similarities in code, suggesting efficient code reusing, code rewriting, etc. These embeddings enable systems to understand the underlying structure, context, and semantics of code, thus facilitating intelligent support for enhancing source code quality.

Code2Vec is a popular approach for code embedding. It was developed by Alon et al. [3] who used deep neural network architecture to create vector representations (code embeddings) of code snippets or functions. Code2Vec was designed to learn distributed representations of source code by considering the context and relationships between code tokens. The model was trained to map snippets of code into fixed-length vectors by extracting paths from the Abstract Syntax Tree (AST) of source code. Although the idea of utilizing AST paths for generating code embeddings is not novel and has been investigated by researchers such as Hu et al. [4], Code2Vec stands out for its innovative approach in utilizing AST paths. Instead of employing the linearization method used by [4], the authors of Code2Vec [3] provide a path-attention network.

While Code2Vec has shown remarkable success in capturing the semantic relationships within source code through its path-attention network, it is essential to acknowledge certain limitations. Notably, Code2Vec exhibits sensitivity to the naming conventions of variables, where minor changes, such as typos or alternative designations, can significantly impact the model's predictions. This sensitivity arises from the model's training on well-structured and best-practice-following code repositories, such as top-rated GitHub projects. In scenarios like genetic programming or diverse coding conventions, where variable names may exhibit more variations, the model's robustness is notably compromised.

In this study, we evaluate the performance of the Code2Vec deep learning model by introducing a pre-processing step consisting of anonymizing method variable names. The primary goal is to improve the Code2Vec model's ability to capture the semantic properties of source code and promote a deeper comprehension of code structures. In the following sections of this paper, we present an analysis that encompasses our methodology, experimental design, and the outcomes derived from our approach.

2 Method

2.1 Experiments Pipeline

This experimentation pipeline outlines a thorough evaluation of the Code2Vec model's performance through a series of steps. Starting with a set of test data extracted from the Java-Small dataset, the pipeline goes through key phases including path context extraction, Code2Vec model evaluation, and result analysis. A distinctive feature of this pipeline is the incorporation of an anonymization step for method variable names, challenging the model's sensitivity to naming variations (see Fig. 1).

2.2 Code2Vec Model

Alon et al. [3] have recently proposed Code2Vec, an interesting approach able to learn distributed representation of code similar to Word2Vec [1]. They proposed a neural network architecture based on attention mechanism in order to predict semantic properties of a given code snippet.

First, the snippet code is transformed to an Abstract Syntax Tree (AST), and then translated into a collection of path-contexts. Path-contexts are established from each leaf to another leaf. The starting and ending nodes and the path between them are represented by vectors, and their values are learned and concatenated into a path context vector (PCV). During training, the model's attention mechanism learns how to aggregate these PCVs. Attention mechanisms are used to assign different weights to different path-contexts, focusing more on specific PCVs (i.e., parts of the code). This allows the model to identify highly predictive parts of the AST (e.g., certain variable names) and ignore those that are not predictive. It has been shown to distinguish small variations between similar Java methods [5] (see Fig. 2).

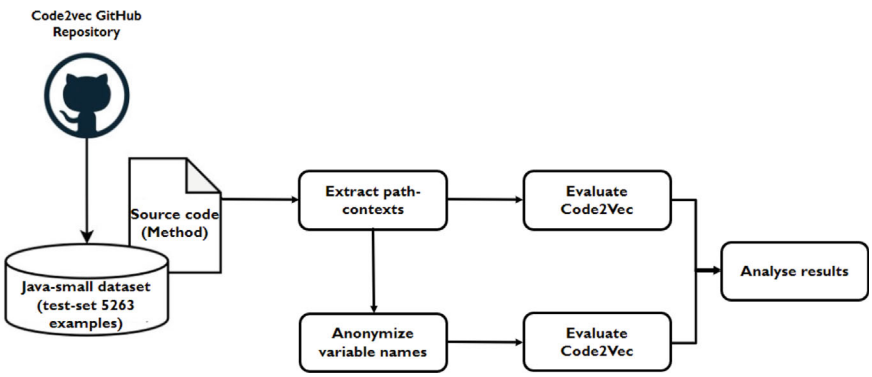


Fig. 1 Overview of the experimentation pipeline

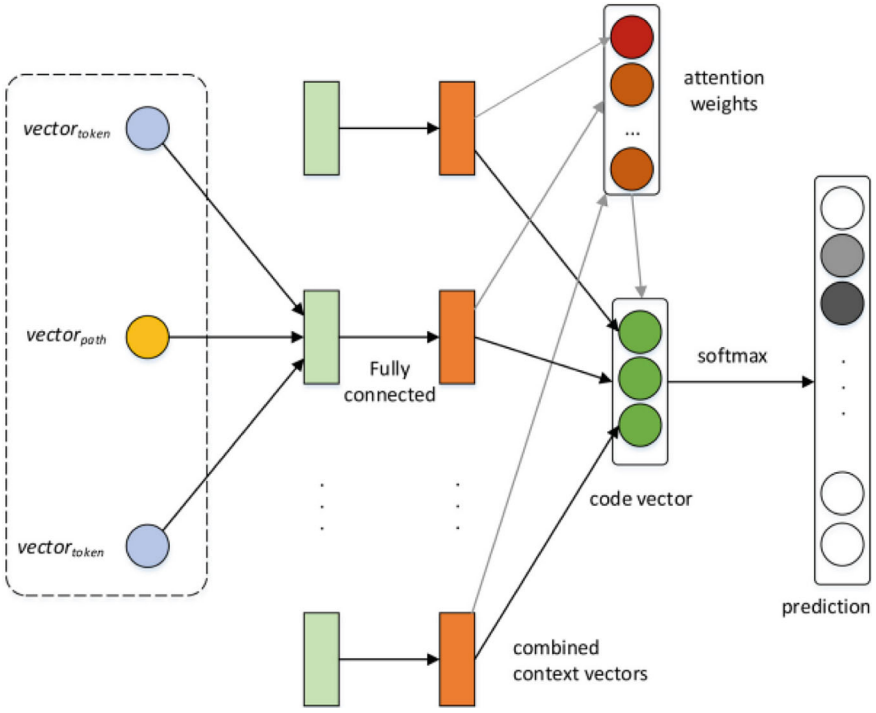


Fig. 2 Code2Vec model architecture [6]

To briefly describe the methodology, it is useful to employ the nomenclature from the original Code2Vec paper [3]:

- **AST Path**: It is a directed sequence of nodes within the AST structure. This sequence connects two specific terminal nodes, indicating the traversal direction (upwards or downwards) from one node to the other.
- **Path-context**: It is defined as a triplet (x_s, p, x_t). Here, p represents the AST path while x_s and x_t denote the values associated with the starting and the ending terminal nodes of the path respectively.
  Example: The following is a possible path-context that represents the statement "x = 7;": $\left(\text{x,(NameExpr} \uparrow \text{AssignExpr} \downarrow \text{IntegerLiteralExpr), 7}\right)$

In the example above, $\uparrow$ indicates movement in the direction of the AST's root, whereas $\downarrow$ indicates movement in the direction of its leaves.

Code2Vec operates under the principle that a code snippet can be viewed as a collection of "path contexts". Each path context represents a specific route within the code's structure. These paths are encoded as vectors, capturing both their semantic meaning (what they do) and their relative significance. It's crucial to note that some path contexts might be common across various methods (e.g., method definitions).

These ubiquitous contexts hold minimal unique information and consequently warrant less attention. The primary challenge lies in consolidating information from all these path contexts into a single, informative representation of the entire code snippet. A simplistic approach might involve selecting the most crucial path context or averaging them all. However, this wouldn't capture the complete picture. Not all paths hold equal weight, and ideally, we aim to utilize all the information. Code2Vec tackles this by assigning "attention scores" to each path context. These scores indicate the level of focus the model should allocate to each individual path. The attention scores are learned concurrently with the actual code representations (embeddings) during the training process.

3 Experiments

During our experiments, we used the official implementation of Code2Vec available on the authors' website.[1] We have noticed that labels prediction of Code2Vec model is very sensitive to variables names in the code snippet. Even minor changes in variable names (e.g., typos or alternative naming) significantly affect the model's prediction, impacting the model's accuracy and reliability. Figure 3 displays two methods with modified variable names (due to an alternative name in the case of the first method and a typo in the case of the second) and the difference in the predicted method name. Minor changes in variable names lead to significantly different predictions for a classifier trained for method name prediction. For instance, on the right, when the "e" is omitted in "done," the model struggles to generate a correct prediction. Additionally, on the left, renaming the variable "factorial" to "total" in a factorial function results in an incorrect prediction. This highlights the model's heavy reliance on variable names for making predictions. This sensitivity is due to the fact that the model is trained on top-rated GitHub projects, where variable names follow best practice and are therefore highly predictive. However, when confronted with scenarios outside this controlled environment, such as genetic programming or code exhibiting diverse naming conventions, the model's robustness is noticeably compromised (see Fig. 3).

We also compare the code embeddings obtained for each of the two versions of the two methods by measuring cosine similarity and Euclidean distance between the vectors. The embeddings for the 'factorial' example in Fig. 3 have a similarity of **0.598** and a Euclidean distance of **9.18**, while the embeddings for the '*done*' method are even more dissimilar, with a cosine similarity of **0.311** and a distance of **12.85**.

These results highlight the detrimental impact of variable name modifications on the precision of Code2Vec representations. In both scenarios, a simple change leads to misleading embeddings and entirely inaccurate output predictions.

In the case of GP, if variable names are present, they are often formed simply by an identical radical followed by a number, for example, registers R0 to Rn in

[1] Code2Vec website: https://code2vec.org/.

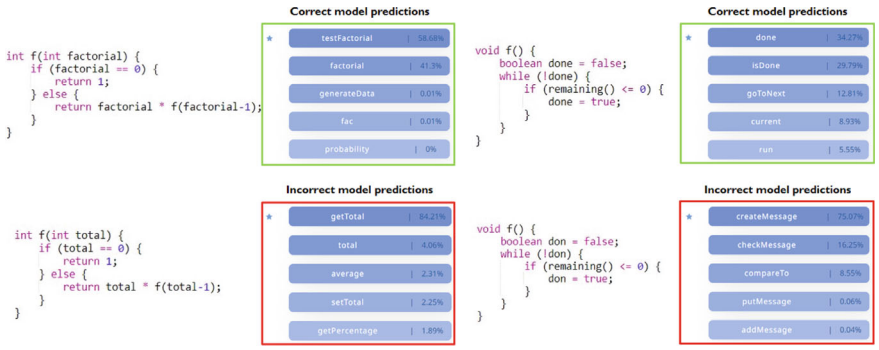


Fig. 3 Significant variations in predictions arising from changes in variable names

Linear Genetic Programming [7]. Moreover, the code is automatically generated, which is quite distinct from code written by humans. Genetic programming evolves and mutates code structures over generations, which often leads to code variations where variable names might change dramatically from one iteration to another. These variations are essential for the GP process, as they help in the exploration of different code solutions.

In our evaluation, we used the pre-trained model available on the authors' Github repository[2] and the java-small dataset collected by Alon et al. [3], based on the dataset of Allamanis et al. [8], with the difference that training, validation, and testing are split by-project rather than by-file. This dataset contains 9 Java projects designated for training, along with 1 project each for validation testing. It comprises a total of approximately 700K code examples [8].

Initially, we extracted the test set (5263 examples) from the java-small dataset and assessed the precision of Code2Vec predictions on this set. This involved measuring the accuracy of the model in predicting method names on the unaltered testing data. Subsequently, we maintained the same test set but introduced a preprocessing step involving anonymization of variable names.

The goal was to assess how robust the model remained when the explicit identification of variable names was obscured. We then re-evaluated the Code2Vec model on the test set with anonymized variable names and recalculated the predictions. This allowed us to compare the precision metrics before and after variable name anonymization, providing insights into the model's sensitivity to naming variations.

The obtained precision metrics were tabulated to clearly illustrate the sensitivity of Code2Vec to variable names. Table 1 showcase the model's performance on the original testing set versus the anonymized variable name set. It highlights a decrease in precision when variable names are anonymized, underscoring the model's dependence on specific variable identifiers for accurate predictions.

[2] Code2vec GitHub Repository: https://github.com/tech-srl/code2vec.

Table 1 Evaluation comparison of 5263 examples based on the small data set from [8] before and after anonymization

Model	Before anonymization	After anonymization
Accuracy	**0.6894**	0.1278
F1-score	**0.6181**	0.1049
Precision	**0.6936**	0.1753
Recall	**0.5574**	0.0980

4 Conclusion

Our investigation will illuminate the future perspectives of Code2Vec, highlighting its current limitations, particularly its sensitivity to variable names. To address this limitation, we propose to include the pre-processing phase involving the anonymization of variable names presented in this paper, followed by retraining the Code2Vec model. This technique aims to shift the model's focus from specific naming details to the underlying semantics of the programs. In other words, instead of relying on naming cues that can vary significantly, the model will be trained to extract more general information about the structure and logic of the code.

The primary goal of anonymizing variable names is to make the Code2Vec model more robust to frequent variations in variable names, especially in contexts such as genetic programming. By eliminating the dependency on specific naming details, we anticipate a significant improvement in the model's ability to generalize beyond the strict naming conventions observed in conventional GitHub projects.

The findings prompt consideration for further research into Code2Vec's learning dynamics with anonymous variables. Given that a program's semantics lie in its operations, and recognizing that specific paths encapsulate these operations, an exploration of whether Code2Vec can be optimized to leverage operations more prominently than variable names could be a promising avenue for future investigation. This avenue holds the potential to enhance Code2Vec's robustness and applicability across diverse coding scenarios.

The results of our experiments show that Code2Vec's performance can be improved by prioritizing operations over variable names, especially when dealing with non-human-generated source code. This suggests that the model's learning process can benefit from a focus on the underlying program semantics, rather than being influenced by variable naming conventions.

In the future, it would be valuable to explore additional techniques and variations of anonymization, assess the impact on diverse code bases, and investigate the model's adaptability to various programming languages. Our work serves as a foundation for future endeavors aimed at refining Code2Vec and similar models, ultimately advancing the state-of-the-art in semantic representation learning for source code analysis.

References

1. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space (2013)
2. Chen, Z., Monperrus, M.: A literature study of embeddings on source code (2019). CoRR http://arxiv.org/abs/1904.03061
3. Alon, U., Zilberstein, M., Levy, O., Yahav, E.: Code2vec: learning distributed representations of code. Proc. ACM Program. Lang. **3**(POPL) (2019)
4. Hu, X., Li, G., Xia, X., Lo, D., Jin, Z.: Deep code comment generation. In: Proceedings—2018 ACM/IEEE 26th International Conference on Program Comprehension. ICPC 2018, pp. 200–210. IEEE, Institute of Electrical and Electronics Engineers (2018)
5. Compton, R., Frank, E., Patros, P., Koay, A.: Embedding java classes with code2vec: improvements from variable obfuscation. In: Proceedings of the 17th International Conference on Mining Software Repositories. MSR'20, pp. 243–253. Association for Computing Machinery, New York, NY, USA (2020)
6. Sun, X., Liu, C., Dong, W., Liu, T.: Improvements to code2vec: generating path vectors using RNN. Comput. Secur. **132**, 103322 (2023)
7. Brameier, M., Banzhaf, W.: Linear Genetic Programming. No. XVI in Genetic and Evolutionary Computation. Springer (2007). https://doi.org/10.1007/978-0-387-31030-5
8. Allamanis, M., Peng, H., Sutton, C.: A convolutional attention network for extreme summarization of source code (2016). CoRR http://arxiv.org/abs/1602.03001

Evaluating the Use of Feature Extraction and Windowing Using Neural Network in EEG-Based Emotion Recognition

Manal Hilali, Abdellah Ezzati, and Said Ben Alla

Abstract Electroencephalogram (EEG) based emotion recognition has gained more attention in recent years due to the emergence of deep learning models, which enabled researchers to dive deeper into the pre-processing, the feature extraction methods, and choosing the right parameters for their model. This study focuses on the windows used to segment an EEG signal and the extracted features while using a deep neural network on the DEAP dataset. The extracted features consist of Deferential Entropy (DE), Power Spectral Density (PSD), Fast Fourier Transform (FFT), and Discrete Wavelet Transform (DWT). Testing the effect of those parameters on the accuracy of a combined neural network architecture. The model used is composed of a Convolutional Neural Network (CNN) and Long Short Term Memory (LSTM) network. The proposed method achieved the highest accuracy for a combination of FFT and a window size of 0.25 s.

Keywords Electroencephalogram (EEG) · Feature extraction · Neural network · DEAP dataset

1 Introduction

Emotion recognition has been a hot topic for many years, either through facial expressions, gestures, or speech, and sometimes the combination of two or more. But those emotions might be subjective, which is why there is another form of emotion recognition through Electroencephalogram (EEG) signals. They can be defined as electrical signals produced during the periods of activity of the brain. These signals are generated by the electrical activity of millions of neurons in the brain as they communicate with each other. They might be used in a variety of applications, including clinical

M. Hilali (✉) · A. Ezzati
Faculty of Sciences and Techniques, Hassan First University, Settat, Morocco
e-mail: m.hilali@uhp.ac.ma

S. Ben Alla
National School for Applied Science, Hassan First University, Berrchid, Morocco

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_11

diagnoses such as epilepsy, sleep disorders, and brain tumors, cognitive neuroscience to study brain functions such as attention, and memory, Brain-Computer Interfaces, and neurofeedback to provide real-time feedback to individuals about their brain activity.

EEG signals are typically measured using a set of electrodes that are placed on the scalp. These electrodes detect the brain's electrical activity and record it as a series of voltage fluctuations over time. The resulting signal is known as an EEG signal.

There are different emotion models in the literature [9], namely the discrete model, defining each emotion by name, and the dimensional model which is used in this study. In the two-dimensional model of valence and arousal, arousal defines the degree of positive or negative emotion, while valence defines the intensity of those emotions.

EEG signals may be categorized into five groups based on how their frequency ranges differ: Delta waves, ranging from 0.5 to 4 Hz, are typically linked to deep sleep. Theta waves, within the 4–8 Hz range, manifest during light sleep. Alpha waves, falling between 8 and 13 Hz, signify a relaxed yet wakeful state. Beta waves, spanning from 13 to 30 Hz, dominate during periods of active thinking. Gamma waves, exceeding 30 Hz, are associated with heightened focus and concentration.

In this paper, related works are discussed in the second section, the proposed methodology is detailed in the third section, and the experimental setup and the dataset are in the fourth. Section five has the result and discussion, then a conclusion in the sixth.

2 Related Work

Most EEG-based emotion detection systems operate by taking into account the complete collected signal from various channels, which is incorrect because emotional states don't stay unchanged over the entire period of signal acquisition. This is why segmenting the signal is essential. Some researchers fixed the widow size [2], while others investigated the effect of different window size [1, 4]. But while considering the baseline signal (3 s) the choice of big window length might not give us the precise effect of the baseline signal on the emotion recognition.

There are many works on the use of different window sizes and the use of different feature extraction methods, AsMap [1] is a model that captures the asymmetry in different brain regions in a 2D vector, they used all five frequencies bands, they extracted the DE features using four different window sizes: 3 s, 6 s, 12 s, 30 s. They achieved an accuracy of 95.45 and 95.21 on valence and arousal respectively.

The authors in Hendy and Isa [2] fixed the window size to 2 s and extracted Discrete Wavelet Transform (DWT) and Fast Fourier Transform (FFT) features using a neural network composed of a CNN and DNN. While [3] used Pearson's Correlation Coefficient (PCC), which displays visuals of the channel correlation of EEG sub-bands, was used to convert the one-dimensional EEG data, they only used 3 frequency bands with a CNN. In Jin et al. [4], PCC characteristics were derived, and diverse

window durations were implemented, specifically 3, 6, and 10 s, employing a Multi-Layer Perceptron network.

In Yang et al. [10, 11] a fixed window size of 1 s was used, the authors in Shen et al. [6] along with the first two works used 4 frequency bands and extracted the DE features while the last one fixed the window size to 0.5 s.

Also in Song et al. [8] a sliding window of one second was used. They demonstrated various feature extraction techniques, those features are PSD, STFT, and others. They designed and build a Multi-Modal Physiological Emotion Database (MPED).

3 Proposed Methodology

3.1 Feature Extraction

For the methodology, the preprocessing started with feature extraction.

We took the signal and decomposed it into pre-trial and trial since many types of research proved the efficiency of taking the first 3 s of de signal into consideration [1, 10]. Those 3 s are essential to extract meaningful information from the signal and they improve the accuracy of the model. Then we used a ranging window to increase the data and test its effect on the prediction performance. The values that the widow took are [0.25, 0.5, 0.75, 1, 1.5]. For each segment, the Butterworth filter is applied to decompose it into 4 frequency bands θ, α, β and γ, since the other frequency band δ is for when the person is in deep sleep, the efficiency of combining those four frequency bands was demonstrated [10]. After that we extract the frequency features (deferential entropy DE, fast Fourier transform FFT, Discrete Wavelet Transform DWT, and power spectral density PSD), average the baseline signal, and subtract it from the trial signal.

From this, we got a feature segment for each frequency band, for the spatial information we map the electrode location into a 2D vector for each feature segment, and the 2D vector of each frequency band is stacked to get finally a 3D vector.

3.1.1 Differential Entropy (DE)

DE [7], which represent the entropy of a random continuous variable, is utilized to assess the complexity associated with such a variable. The minimum description length is also correlated with differential entropy. It can be calculated using the formula:

$$h(X) = -\int_X f(x)\log(f(x))dx \tag{1}$$

where X is a random variable and f(x) is X's probability density function. The differential entropy of the time series X that follows the Gauss distribution, $N(\mu, \sigma^2)$, is defined as

$$h(X) = -\int_{-\infty}^{\infty} \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) e^{-\frac{(x-\mu)^2}{2\sigma^2}} \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}\right) dx$$
$$= \frac{1}{2}\log\left(2\pi e\sigma^2\right) \tag{2}$$

where ⌉ is the Euler constant and σ is the variance of the signal.

3.1.2 Fast Fourier Transform (FFT)

FFT algorithm facilitates the conversion of data between the time domain and the frequency domain.

By splitting the original vector into two pieces, performing an FFT on each piece, and then combining them, the FFT method operates recursively.

Mathematically, for a time-domain signal $x(t)$ and its frequency-domain representation $X(f)$, the FT operates as:

$$X(f) = \int x(t) e^{-ift} \mathrm{dt} \tag{3}$$

The FFT is a rapid method for calculating the Discrete Fourier Transform (DFT), where:

$$X(k) = \sum x(n) W_N^{nk} \tag{4}$$

where $x(n)$ is the signal discrete presentation, $X(k)$ the DFT of the signal often referred to as the spectrum, and $W_N = e^{-i\frac{2\pi}{N}}$.

In the FFT Eq. (4) is disassembled into several shorter transforms, which are later reassembled.

3.1.3 Discrete Wavelet Transform (DWT)

The Wavelet Transform (WT), a method that separates the signal into several frequency spectrums using multi-resolution analysis, is employed. WT also blends the high-frequency and low-frequency spectrums. The efficiency of DWT stems from its utilization of a frequency filter bank. This bank effectively isolates undesired frequencies, allowing the signal to be broken down into multiple hierarchical levels. This hierarchical decomposition enables a detailed analysis of the signal across different frequency bands, providing valuable insights into its underlying

characteristics. The key feature of DWT lies in its ability to finely resolve both frequency and time domains. Essentially, the DWT allows for a granular examination of how frequencies evolve over time, providing a nuanced understanding of signal behavior in different temporal contexts. This level of resolution is crucial for accurately capturing complex signal patterns and variations. The first signal to enter the band-pass filter starts the process. To get the desired outcome, a band-pass filter combines a high-band-pass filter (HPF) and a low-band-pass filter (LPF). This procedure falls under the first level, which has the following two associated coefficients: The first is an Approximation (A), and the second is Detailed (D). A filter that parses and cuts the temporal complexity in half is used in each run to increase the frequency resolution by two times. Use the statistical formulas to obtain statistical features after obtaining those coefficients. In this study, Mean, Standard Deviation (SD), and Variance are the statistical features extracted.

3.1.4 Power Spectral Density (PSD)

The ability to extract EEG features using Power Spectral Density, a measurement of signal power content over frequency, has been demonstrated. Here, the Hamming window and Welch's technique are employed to estimate the PSD.

Mathematically, the Power Spectral Density (PSD) of a continuous signal is computed using the Fourier Transform. The PSD is estimated using a discrete representation of the signal, which involves segmenting the signal and computing, for each of these segments, the Fourier Transform.

Welch's method is a technique for estimating the PSD by dividing the signal into overlapping segments, applying a windowing function (such as the Hamming window), and then averaging the resulting periodograms. This approach helps to reduce variance and improve the accuracy of the PSD estimate.

The Hamming window is a mathematical function that helps in reducing spectral leakage when estimating the PSD.

3.2 Deep Neural Network Architecture

Similarly to the CNN architecture used in Shen et al. [6] and Yang et al. [10]. We employ CNN to capture both the frequency and spatial information from every temporal slice within a sample.

It has one fully connected layer, one max-pooling layer, and four convolutional layers. The 64 feature mappings in the first convolutional layer with a filter size of (5, 5). The following two convolutional layers have 128 and 256 feature maps, respectively, with a filter size of (4, 4). Then 64 feature maps with a filter size of (1, 1) in the fourth convolutional layer.

Zero-padding and the rectified linear units (ReLU) activation function is used for all convolutional layers. To reduce overfitting and improve the robustness of the

network, a max-pooling layer (Pool) of size (2, 2) and a stride of 2 are used after convolutional operations. The Pool layer's outputs are then given as input to a dense layer that consists of 512 units.

To extract the temporal features from CNN outputs, we use an RNN with LSTM cells, since the CNN might not be able to get some more information that can affect the accuracy of emotion prediction because EEG signals contain dynamic content. Hence, the need to use a time series model.

We employ an LSTM layer with 128 memory cells to uncover the inner segment's time dependence. The LSTM layer's output is computed.

4 Experiment Methodology

4.1 DEAP Dataset

Since many EEG-based emotion recognition studies have made extensive use of the DEAP dataset [5], we use it to validate our methodology. After 32 individuals watched 40 one-minute music videos, their EEG signals were captured and included in the dataset. A self-assessment for valence and arousal was required of each individual after seeing the film. Scales for valence and arousal range from 1 to 9, with 1 signifying sadness or calm and 9 signifying happiness or excitement.

A pre-processed form of the DEAP dataset was also made available so that researchers may easily test their suggested emotion recognition techniques. The 32-channel Biosemi ActiveTwo device was used to record the subjects' EEG data following the international 10–20 system.

Prior to the dataset's publishing, EMG and EOG signals were eliminated. The EEG signals were downsampled to 128 Hz and passed through a 4–45 Hz filter to cancel noise.

Each trial's data was divided into 60 s of experimental signals (recorded while viewing the video) and 3 s of baseline signals (recorded in a relaxed state).

Data and labels are two arrays of each participant's EEG data in the pre-processed version of the DEAP dataset shown in Table 1. It contains 40 trials * 40 channels * 8064 data points, we used only 32 channels and divided the data into baseline 3 s and trial 60 s, which corresponds to 384 and 7680 data points respectively.

Table 1 The summary of EEG data in DEAP dataset

Array name	Array shape	Array contents
Data	40 × 40 × 8064	Video/trial × channel × data
Labels	40 × 4	Video/trial × label (valence, arousal, dominance, liking)

4.2 Experimental Setup

Adam is used to training the model, and the parameters batch size, learning rate, and epochs are fixed to 128, 0.001, and 100 respectively. Keep in mind that the test set was used to optimize all of these training hyper-parameters. The model is developed using Keras, a Google TensorFlow extension, and trained on the PAPERSPACE platform Gradient module using the already set-up Jupiter notebook with GPU.

The mean prediction accuracy of all algorithms for EEG emotion recognition along with its standard deviation are evaluated by applying fivefold cross-validation on each subject; they represent the individual performance of the subject. The overall effectiveness of the methodology is gauged through the mean Accuracy and standard deviation computed across all subjects.

5 Results and Discussion

The CNN network under consideration accepts 3D vectors with size (h, w, d) as inputs. The temporal information extracted is influenced by the window size. Consequently, we investigate how the length of the EEG segment (T) affects the accuracy of the recognition process. Next, we assess the effectiveness of the different extracted features. Our comparison with other structures comes last.

We study the segment length T ranges in [0.25, 0.5, 0.75, 1, 1.5], knowing that the window size impacts the information of an EEG segment. In addition, we fix the height and the width of the 3D vector to 8 and 9, and d, the number of frequency bands to 4. The Performances of various segment lengths on EEG signals from DEAP datasets are summarized in Fig. 1.

We can see that the overall accuracy is better when the window size gets smaller, even though when T = 1 s it gave better accuracy than when T = 0.75 s generally, except for DE. The best accuracy is given by FFT features, followed by DE, then DWT, and PSD comes last for a window size T = 0.25 s.

We can infer two things from the findings. Since it produces the highest results with FFT features, T = 0.25 seems to be the ideal window size for predicting EEG emotions, it gave the best accuracy for all the methods used. The average ACCs and STDs for the valence and arousal categorization tasks are $97.40 \pm 0.82\%$, and $97.35 \pm 0.84\%$, respectively when we use FFT extracted features, as shown in Table 2, which summarize the accuracy for different window sizes using FFT extracted features.

Since the best result is obtained when FFT features are used, Table 2 shows that the accuracy of arousal is slightly better than the valence accuracy which might be due to data imbalance.

To further see the effect of FFT with window size T = 0.25 s, we will present the accuracy of valence and arousal for each one of the 32 subjects in Fig. 2.

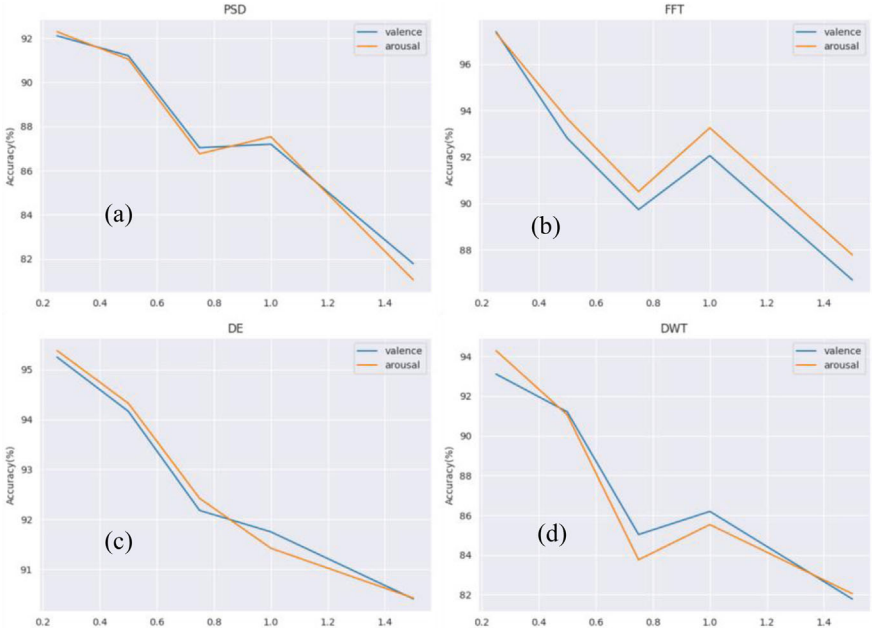


Fig. 1 The accuracy for different window sizes [0.25, 0.5, 0.75, 1, 1.5] for **a** PSD features **b** FFT features **c** DE features **d** DWT features

Table 2 Valence and arousal accuracy for FFT features using different window sizes [0.25, 0.5, 0.75, 1, 1.5]

Windows (s)	Valence (%)	Arousal (%)
0.25	97.40	97.35
0.5	92.81	93.65
0.75	89.74	90.51
1	92.06	92.29
1.5	86.71	87.79

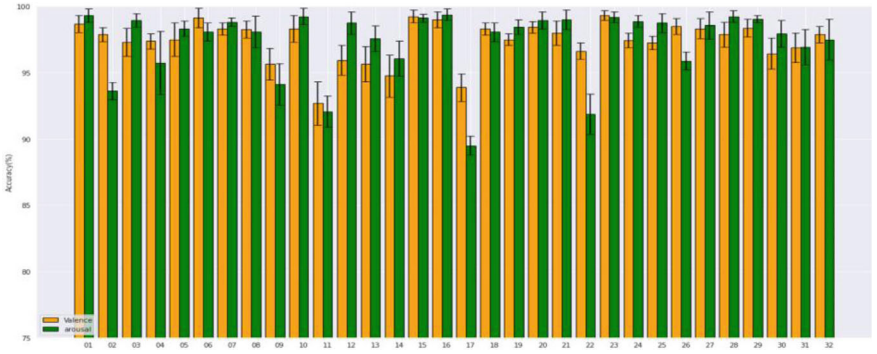


Fig. 2 Individualized Accuracy Scores for Valence and Arousal across 32 Subjects in the DEAP

Table 3 Comparison with state-of-the-art models

Methods	Window (s)	Feature	Valence (%)	Arousal (%)
4DCRNN [4]	0.5	DE	94.22	94.58
CCNN [5]	0.5	DE	89.80	90.50
PCRNN [8]	0.5	DE	90.26	90.98
HEC [2]	2	FFT DWT	96.84	97.18
AsMAP [1]	3	DE	95.45	95.21
OURS	0.25	DE	95.25	95.39
OURS	0.25	DWT	93.10	94.38
OURS	0.25	PSD	92.11	92.29
OURS	0.25	FFT	**97.40**	**97.35**

The overall accuracy of 99% was achieved for most of the subjects, while we observe a slightly lower accuracy in arousal for #2, in arousal and accuracy for #9, and also for #11 in arousal and accuracy, and only arousal #17 and #22.

To show the effective performance of our proposed method, we compare our methodology to the state-of-the-art models based on the window used, then the extracted features comparing the accuracy of predicting the arousal and valence emotions, the details are summarized in Table 3.

From Table 3, we could see that our methodology proved its effectiveness compared to existing methods. While using a window size of 0.25 s which corresponds to a segment length of 32, our method with PSD features, which gave us lesser accuracy, is still better compared to two of the existing methods, namely CCNN and PCRNN with an improvement of 1.31% and 1.85% in arousal and valence respectively for PCRNN, and an accuracy improvement of 1.79% and 2.31% compared to CCNN. DWT features also when combined with a window size of 0.25 s gave us a clear improvement compared to the two models cited above.

When using the DE features combined with a 0.25 s window size, it gave us a 0.18% improvement in arousal compared to AsMAP and a close accuracy of −0.2% in valence, while compared to 4DCRNN an increase of 0.81% and 1.03% in arousal and valence respectively.

As for the FFT features, our method outperforms the other models by far and exceeds HEC by an accuracy of 0.17% and 0.56% for arousal and valence respectively.

6 Conclusion

In this work, we worked on testing the effect of the window length, ranging from 0.25 to 1.5 s, on the accuracy of the model, composed of a CNN and an LSTM network, when we consider the baseline signals. All while using different feature extraction,

namely DE, FFT, PSD, and DWT. And as shown in the results above, the adequate window size is 0.25 s when combined with FFT-extracted features, it obtained an outstanding accuracy of $97.40 \pm 0.82\%$, and $97.35 \pm 0.84\%$ for valence and arousal respectively compared to the state-of-the-art methods.

References

1. Ahmed, M.Z.I., Sinha, N., Phadikar, S., Ghaderpour, E.: Automated feature extraction on AsMap for emotion classification using EEG. Sensors **22** (2022).https://doi.org/10.3390/s22062346
2. Hendy, Isa, S.M.: Human emotion classification from electroencephalogram signal using feature extraction methods and deep learning. ICIC Express Lett. **16**, 1157–1167 (2022). https://doi.org/10.24507/icicel.16.11.1157
3. Islam, Md.R., Islam, Md.M., Rahman, Md.M., Mondal, C., Singha, S.K., Ahmad, M., Awal, A., Islam, Md.S., Moni, M.A.: EEG channel correlation based model for emotion recognition. Comput. Biol. Med. **136**, 104757 (2021). https://doi.org/10.1016/j.compbiomed.2021.104757
4. Jin, L., Chang, J., Kim, E.: EEG-based user identification using channel-wise features. In: Palaiahnakote, S., Sanniti di Baja, G., Wang, L., Yan, W.Q. (eds.) Pattern Recognition. Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 750–762 (2020). https://doi.org/10.1007/978-3-030-41299-9_58
5. Koelstra, S., Muhl, C., Soleymani, M., Lee, J.-S., Yazdani, A., Ebrahimi, T., Pun, T., Nijholt, A., Patras, I.: DEAP: A database for emotion analysis; using physiological signals. IEEE Trans. Affect. Comput. **3**, 18–31 (2012). https://doi.org/10.1109/T-AFFC.2011.15
6. Shen, F., Dai, G., Lin, G., Zhang, J., Kong, W., Zeng, H.: EEG-based emotion recognition using 4D convolutional recurrent neural network. Cogn. Neurodyn. **14**, 815–828 (2020). https://doi.org/10.1007/s11571-020-09634-1
7. Shi, L.-C., Jiao, Y.-Y., Lu, B.-L.: Differential entropy feature for EEG-based vigilance estimation. In: 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). Presented at the 2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 6627–6630 (2013). https://doi.org/10.1109/EMBC.2013.6611075
8. Song, T., Zheng, W., Lu, C., Zong, Y., Zhang, X., Cui, Z.: MPED: a multi-modal physiological emotion database for discrete emotion recognition. IEEE Access **7**, 12177–12191 (2019). https://doi.org/10.1109/ACCESS.2019.2891579
9. Wang, J., Wang, M.: Review of the emotional feature extraction and classification using EEG signals. Cogn. Robot. **1**, 29–40 (2021). https://doi.org/10.1016/j.cogr.2021.04.001
10. Yang, Y., Wu, Q., Fu, Y., Chen, X.: Continuous convolutional neural network with 3D input for EEG-based emotion recognition. In: Cheng, L., Leung, A.C.S., Ozawa, S. (eds.) Neural Information Processing. Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 433–443 (2018). https://doi.org/10.1007/978-3-030-04239-4_39
11. Yang, Y., Wu, Q., Qiu, M., Wang, Y., Chen, X.: Emotion recognition from multi-channel EEG through parallel convolutional recurrent neural network. In: 2018 International Joint Conference on Neural Networks (IJCNN). Presented at the 2018 International Joint Conference on Neural Networks (IJCNN), pp. 1–7 (2018). https://doi.org/10.1109/IJCNN.2018.8489331

Arrhythmia Detection in Single-Lead Heartbeat Using ECG Residual Architecture

Nadia Berrahou, Hatim Jamali, Abdelmajid El Alami, Abderrahim Mesbah, Rachid El Alami, Hassan Qjidaa, and Aissam Berrahou

Abstract Cardiovascular diseases (CVD) stand as one of the gravest threats to human life. The electrocardiogram (ECG) emerges as a highly effective tool for CVD detection. These systems extract important information from a patient's cardiac conditions and provide it to specialists. However, accurately and quickly diagnosing arrhythmia in ECG classification systems is challenging due to factors such as noise, individual variability in the morphology of heartbeats, and data imbalances. Achieving accurate and timely diagnoses of CVDs is vital for effective treatment and patient recovery. In this paper, we introduce a new CNN model designed for classifying heartbeat segments extracted from single-lead ECG. Our approach involves utilizing discrete wavelet transformation with the Sym7 mother wavelet and employing the SMOTE oversampling algorithm as a pre-processing step. The feature extraction and classification are performed by a 1D Residual Convolutional Neural Network. To assess the effectiveness of our proposed model, we conducted both training and testing phases using the MIT-BIH dataset, to identify five arrhythmia categories. We designed our experiment with two distinct scenarios: intra-patient and inter-patient. Our results demonstrate outstanding average classification accuracy of 99.53% for intra-patient and 97.87% for inter-patient, surpassing the performance results reported in recent literature on similar studies. Furthermore, our proposed model exhibits excellent performance in other evaluation metrics, including precision and sensitivity, indicating its success in accurately classifying arrhythmias.

Keywords Cardiovascular diseases · Convolutional neural network · Electrocardiogram signals · MIT-BIH arrhythmia dataset · AAMI

N. Berrahou (✉) · A. E. Alami · R. E. Alami · H. Qjidaa
Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: nadia.berrahou@gmail.com

H. Jamali · A. Mesbah · A. Berrahou
Mohammed V University, Rabat, Morocco

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_12

1 Introduction

Cardiovascular Disease (CVD) stands as a major global health concern, responsible for around 18 million deaths annually, accounting for 32% of total deaths according to the 2019 World Health Organization (WHO) report [1]. Among the various forms of CVD, arrhythmia, characterized by irregular heart rhythms, is a significant representative. Arrhythmias encompass diverse types, including atrial fibrillation, premature contractions, ventricular fibrillation, and tachycardia.

Electrocardiogram (ECG) signals are indispensable in clinical settings for evaluating cardiac function. They provide a temporal representation of the heart's electrical activity. To discern various arrhythmias, the Association for the Advancement of Medical Instrumentation (AAMI) has established five primary heartbeat classes (N, S, V, Q, and F) [2]. The diagnosis of these critical conditions is a complex and time-consuming process, often challenging even for medical professionals. Moreover, the extensive examination of multiple variables adds to the complexity and cost of the diagnostic process. Hence, there's a compelling need to develop automated techniques for accurate cardiovascular disease detection.

Automated techniques for classifying heartbeats within ECG signals typically rely on feature extraction and classification methodologies [3]. These methods often utilize features such as heartbeat morphology and RR intervals. A variety of algorithms, including Artificial Neural Networks (ANNs), Support Vector Machines (SVMs), multiview-based learning, and Linear Discriminants (LDs) have been employed for classification. However, practical implementation can be challenging due to the presence of noise and physiological artifacts in ECG signals, as well as the variability between individuals and within the same individual over time [4]. Deep learning models have the advantage of automatically learning and extracting features directly from raw ECG data. Studies have shown that deep learning-based methods excel at handling variations between patients in ECG classification [5, 6], generally outperforming classical methods. Nevertheless, deep learning in this context presents certain challenges.

One challenge is the potential for overfitting, especially with deep models that have many layers [6]. Overfitting arises when a model becomes overly attuned to the training data, leading to diminished performance when presented with new, unseen data. This emphasizes the importance of careful model design and regularization to avoid overfitting.

Another challenge is the computational resources needed by deep learning models. Larger models with more layers often have longer processing times and high memory requirements, making them less efficient for long-term ECG monitoring. Balancing model complexity and resource efficiency is crucial for practical applications.

Additionally, addressing the class imbalance problem is critical in ECG classification. Imbalanced data, where some classes have significantly fewer instances than others, can bias classification algorithms toward the majority classes. This imbalance often leads to reduced accuracy in detecting minority classes [7]. Techniques such as oversampling, under-sampling, cost-sensitive learning, ensemble methods, and specialized loss functions are used to mitigate class imbalance [8].

Despite significant progress, there's room for improvement in ECG classification using deep learning. Performance needs to be enhanced to meet modern diagnostic demands. Research should also move toward inter-patient schemes for more realistic evaluation, and standardization of cardiac arrhythmia types and evaluation methods would improve reproducibility and comparison of experiments [9].

Motivated by these challenges and the goal of improving classification performance, this paper introduces a new CNN architecture for ECG classification by using Discrete Wavelet Transform (DWT) and Convolutional Neural Network (CNN).

The structure of the paper is as follows: Sect. 2 provides a comprehensive literature review. In Sect. 3, the materials and methods employed in the study are outlined. Section 4 introduces our novel CNN approach for ECG classification. Section 5 presents experimental findings, including comparisons with previous studies regarding classification performance. Finally, Sect. 6 offers conclusions and proposes future research directions.

2 Related Works

2.1 ECG Signal

The electrocardiogram (ECG) is a critical diagnostic tool used to visualize the heart's electrical activity during its operation. By recording electrical signals produced by the heart on the body's surface, the ECG aids in detecting and diagnosing various heart-related conditions. Cardiologists rely on ECG recordings to identify cardiac arrhythmias, assess structural changes in the heart's chambers, and diagnose ailments such as ischemia and myocardial infarction. The ECG signal consists of distinct waves P, Q, R, S, and T, each representing specific phases of the heart's electrical cycle. Key features of the ECG morphology include the P wave, QRS complex, and T wave, corresponding to different stages of the cardiac electrical impulse's generation and propagation.

2.2 Literature Review in ECG Signal Analysis

Over the decades, researchers worldwide have made substantial progress in advancing the field of ECG signal classification, using a wide array of methodologies, ranging from traditional techniques to more advanced machine learning and deep learning methods.

Early ECG signal classification methods were reliant on a sequential process involving three main steps: signal preprocessing, feature extraction, and classifier design. The primary objective of preprocessing was to eliminate various forms of noise, including motion artifacts and power line interference, from ECG recordings. A multitude of denoising techniques were explored in the literature, such as

low-pass filters, adaptive filters, and filter banks [10]. After preprocessing, important fiducial points were correctly identified and extracted, which provided useful morphological information about the heartbeats, such as the P wave, R peak, T wave, and QRS complex. Several algorithms and techniques were developed to precisely locate these fiducial points within the ECG signal [11]. The subsequent step includes extracting features from ECG segments. Many algorithms for classifying ECG beats have relied on manually engineered feature extraction techniques. However, the introduction of machine learning, particularly supervised classification techniques, marked a profound transformation in ECG signal classification. For example, Elhaj et al. [12] integrated Principal Component Analysis (PCA) with Discrete Wavelet Transform (DWT) coefficients in their method for classifying ECG signals. Furthermore, they employed both a support vector machine and a neural network for classification, resulting in an impressive intra-patient classification accuracy of 98.91%.

Shi et al. [13] proposed a hierarchical classification approach using weighted extreme gradient boosting (XGBoost) and recursive feature reduction, leading to an inter-patient classification accuracy of 92.1%. Chen et al. [14] integrated weighted RR-interval features with projected ECG features, demonstrating high classification performance in the intra-patient evaluation paradigm but encountering reduced sensitivity and precision in the inter-patient evaluation paradigm.

Despite achieving high classification rates, many of these methods heavily relied on the quality of features extracted from the ECG signals. Numerous frameworks treated ECG signals as sequences of stochastic patterns, necessitating intricate feature extraction processes and elevated sampling rates. However, the robustness of these techniques was limited due to significant intra-class variation observed in ECG signals.

The advent of deep learning marked a significant turning point in ECG signal classification. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) showcased their ability to autonomously extract discriminative features. CNN, particularly, became widely adopted in ECG classification due to its ability to automatically capture essential patterns and features.

Within the intra-patient paradigm of ECG beat classification, various approaches were proposed. For instance, Acharya et al. [6] introduced a nine-layer convolutional neural network (CNN) for the classification of five different categories of ECG beats using the MIT-BIH arrhythmia dataset. They tackled the challenge of class imbalance among these categories by employing a data augmentation technique, which involved generating synthetic heartbeats by modifying the statistical properties of the original data. This approach led to an impressive overall classification accuracy of 94.03%. Other approaches for heartbeat classification, including CNN-based models, were proposed in Refs. [15, 16]. Ubeyli [17] introduced a classifier based on recurrent neural networks (RNN) with an eigenvector method, achieving an average accuracy of 98.06% in detecting four distinct classes of arrhythmia.

Long Short-Term Memory Networks (LSTMs) have emerged as potent tools for distinguishing between beat classes in arrhythmia detection. Studies cited in [18, 19] developed LSTM-based models showcasing robust recognition performance across

five beat categories, while reducing computational overhead, demonstrating LSTM's effectiveness in capturing temporal dependencies in ECG data. Furthermore, a hybrid model combining Convolutional Neural Networks (CNNs) and LSTMs, introduced in [20], leverages both temporal and spatial information in ECG signals. Significantly, it employs ECG segments of varying lengths, thereby improving adaptability in arrhythmia detection.

Within the context of inter-patient beat classification, various approaches were developed to classify heartbeats from different patients. For example, Sellami et al. [5] designed a powerful deep Convolutional Neural Network for arrhythmia classification without denoising the ECG signals. Subsequently, Zhang et al. [21] introduced a deep learning model for inter-patient heartbeat classification, utilizing a convolutional neural network (CNN) with an adversarial approach. In this model, separate channels are employed for processing ECG heartbeat segments and normalized RR intervals, allowing the network to effectively learn and classify heartbeats from different patients.

Alternatively, Wang et al. [22] presented an inter-patient ECG model using CNNs and Continuous Wavelet Transform (CWT). A 2D CNN was then employed alongside CWT-generated time-frequency scalograms of ECG segments and RR intervals for beat classification. Nevertheless, the CWT preprocessing step introduced extra computational overhead to the classifier, which could potentially constrain its suitability for edge inference scenarios. Despite the significant progress made in ECG classification, there remain opportunities for further performance improvement. To address these limitations, the proposed study introduces a novel ECG classification model based on a 1D convolutional neural network (CNN). We rigorously evaluate this model using intra-patient and inter-patient paradigms to assess its effectiveness and robustness. Our model showcases superior performance compared to existing literature. Leveraging the advantages of deep learning, this approach enhances the accuracy and efficiency of ECG signal classification.

3 Materials and Methods

Figure 1 illustrates the flowchart for classifying ECG signals. Initially, we preprocess the ECG record to create input data. Subsequently, this data is fed into our proposed CNN model, which extracts pertinent features, leading to the ultimate classification outcome.

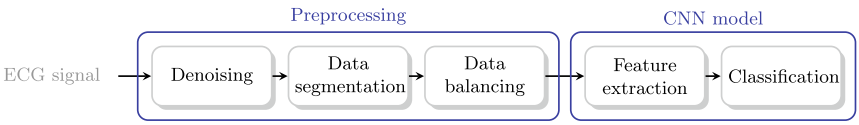


Fig. 1 Flowchart of automatic classification arrhythmia

3.1 MIT-BIH Arrhythmia Dataset

In this paper, we conducted our research using the PhysioNet MIT-BIH Arrhythmia dataset [23] to assess the performance of our proposed method. The MIT-BIH dataset, provided by the Massachusetts Institute of Technology and conforming to international standards, contains annotated ECG data contributed by multiple experts. This dataset has been widely adopted by the academic community for QRS detection and classification research. It is particularly recognized for its suitability in evaluating ventricular arrhythmia detection systems, as endorsed by (AAMI) [2].

The MIT-BIH dataset consists of 48 ECG recordings, with each lasting 30 min and sampled at a frequency of 360 Hz. These recordings include data from two ECG leads: lead II and lead V1. It's worth noting that lead II is commonly utilized for heartbeat detection in the research literature, and consistently, we employed ECG lead II in all our experiments.

The heartbeats within the MIT-BIH dataset are categorized into 15 distinct classes, which can be further grouped into five categories as outlined in Table 1. As per the recommended guidelines from AAMI, records 102, 104, 107, and 217 were excluded from our analysis due to insufficient signal quality for diagnosing cardiac diseases.

Table 1 Heartbeat categories in the MIT-BIH dataset according to AAMI standards

Groups	ECG class	Number
(N)	Normal beat (N)	90124
	Left-bundle-branchblock (L)	
	Right-bundle-branchblock (R)	
	Atrial escape (e)	
	Nodal escape (j)	
(S)	Atrial premature (A)	2781
	Aberrant atrial premature (a)	
	Nodal (Junctional) premature (J)	
	Supra-ventricular premature (S)	
(V)	Premature ventricular contraction (V)	7009
	Ventricular escape (E)	
(F)	Fusion of ventricular and normal (F)	803
(Q)	Paced (/)	15
	Fusion of paced and normal (f)	
	Unclassifiable (Q)	

Table 2 Distribution of the inter-patient paradigm in the MIT-BIH arrhythmia dataset

Datasets	N	S	V	F	Q	Records
DS1	45866	944	3788	415	8	101, 106, 108, 109, 112, 114, 115, 116, 118, 119, 122, 124, 201, 203, 205, 207, 208, 209, 215, 220, 223, 230
DS2	44258	1837	3221	388	7	100, 103, 105, 111, 113, 117, 121, 123, 200, 202, 210, 212, 213, 214, 219, 221, 222, 228, 231, 232, 233, 234

3.2 Paradigms

To evaluate the proposed model, we employed two main paradigms: the intra-patient and inter-patient paradigms. In the intra-patient paradigm, the dataset is divided into training and test subsets based on individual patient data. This ensures that data from each patient is exclusively present in either the training or test set, enabling rigorous testing on unseen patient data.

Contrarily, the inter-patient paradigm involved the division of the MIT-BIH arrhythmia dataset into two discrete subsets, DS1 and DS2, as described in [3]. Each dataset consists of 22 records and maintains a consistent distribution of beat types. Specifically, DS1 serves as the training dataset, while DS2 is reserved for evaluating the model's performance. Table 2 summarizes the inter-patient paradigm distribution in the MIT-BIH arrhythmia dataset.

3.3 Preprocessing

Ensuring the proposed model performs at its best requires careful preprocessing of the raw data. This process involves three key steps: ECG denoising, heartbeat segmentation, and addressing class imbalance. Each step is vital for improving the data quality and setting a solid foundation for further analysis.

ECG Denoising

The raw ECG signal sourced from the MIT-BIH dataset often contains various types of interference, collectively referred to as noise. Noise within the raw ECG signal can pose challenges in extracting meaningful information. To address this issue, a denoising process is applied to the ECG signal, combining a bandpass filter before segmentation and Symlet wavelet transform after the segmentation step.

- Bandpass Filter: To mitigate unwanted noise in the input signal, we employ a bandpass filter that operates in two stages to establish a passband ranging from 5 to 15 Hz. Initially, the signal traverses a low-pass filter with a cutoff frequency of 11 Hz, followed by processing through a high-pass filter with a cutoff frequency of 5 Hz. This sequential operation effectively achieves the desired passband and aids in noise reduction.

The recursive equation governing the low-pass filter is expressed as:

$$y(nT) = 2y(nT - T) - y(nT - 2T) + x(nT) - 2x(nT - 6T) + x(nT - 12T) \quad (1)$$

Similarly, the high-pass filter follows the recursive equation:

$$y(nT) = 32x(nT - 16T) - y(nT - T) - x(nT) + x(nT - 32T) \quad (2)$$

Application of this filter in the frequency domain effectively eliminates frequencies smaller than 5 Hz and greater than 30 Hz, thereby smoothing the signal and centering it around zero.

- Symlet Wavelet Transform: We utilize the Symlet 7 (Sym7) mother wavelet within the discrete wavelet transform (DWT) framework to bolster denoising efforts. Sym7 is selected due to its intricate nature and its resemblance to ECG signals, rendering it a fitting choice for signal preprocessing tasks.
  The signal undergoes decomposition into seven levels, with particular emphasis on retaining wavelet coefficients from the 3rd to the 6th level. These coefficients are then leveraged for reconstructing the denoised signal, thereby enhancing its quality and fidelity for subsequent analysis.

Data segmentation

Before conducting ECG classification, a critical phase involves the extraction of individual heartbeats from the ECG signal. This process typically necessitates the accurate detection of QRS waves and the localization of key points within the heartbeats.

In our study, we tackled this segmentation task by utilizing the positional data of R-peaks annotated by the MIT-BIH arrhythmia dataset. Each isolated heartbeat comprised 400 samples, with its center precisely aligned around the R-peak. These 400 sample points effectively encapsulated the most pertinent waveforms within the heartbeats, ensuring that essential features were captured for subsequent analysis and classification.

Data balancing

The MIT-BIH arrhythmia dataset displays a notable class imbalance, as indicated in Table 1. Specifically, normal heartbeats make up 89% of the data, which can potentially lead to challenges like overfitting and a bias towards the normal class during model training. To tackle this issue, our model employs the Synthetic-Minority Oversampling Technique (SMOTE) [24]. SMOTE is a technique designed to address class imbalance by generating synthetic examples for the minority class, thereby increasing its representation in the dataset while reducing instances from the majority class. In this paper, we oversampled the classes (S, V, F, and Q) while undersampling the class N. This resulted in a balanced distribution where each class comprised 20% of the overall training dataset.

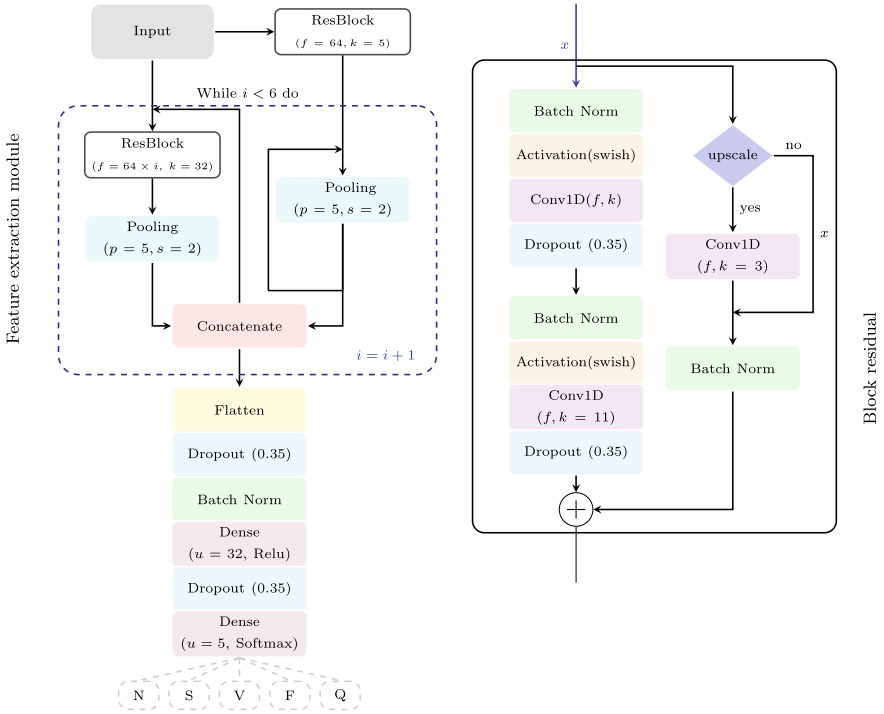


Fig. 2 Architecture of the proposed model

4 Architecture of Our Proposed Model

A Convolutional Neural Network (CNN) is primarily developed for image classification tasks. It's traditionally utilized with two-dimensional spatial input data, like images, where it excels at learning hierarchical features. However, CNNs can also be adeptly applied to process one-dimensional sequential data. Leveraging insights into the characteristics of ECG signals and CNN capabilities, we introduce a CNN model, depicted in Fig. 2.

The architecture of our CNN model consists of two main parts: the feature extraction and the classification. The first one consists of batch normalization, convolution, activation, and pooling layers, while the second part consists of flatten, fully connected, and SoftMax layers. To enhance the model's ability to capture global context information, we incorporated a residual connection method inspired by deep residual networks.

All heartbeat segment of the ECG signal undergoes two distinct pathways for feature extraction. On one side, it passes through a 1D Resblock before entering a feature extraction module enclosed in dashed lines. On the other side, the input signal

directly feeds into the same feature extraction module. This ingenious design aims to effectively capture various relevant features of the input signal.

Each 1D Resblock consists of three key elements: three batch normalization (BN) operations, two Swish activation functions, three convolutional layers, and a residual connection, as depicted in Fig. 2. It is structured in a full pre-activation manner (The pre-activation architecture is implemented by moving BN and Swish activation function before the convolution operation). The batch normalization layer plays a crucial role in normalizing feature distributions of the data at different levels, ensuring stable loss and gradient values. Concurrently, the activation layer introduces non-linearity into the model, enhancing its classification capabilities. To prevent overfitting and reduce the model's parameter count, a dropout layer with a dropout rate of 0.35 follows each 1D convolutional layer.

In the configuration of the 1D Resblock, the number of filters for all convolutional layers is set to 64. The kernel for the three convolutional layers is established as 5, 11, and 3, respectively. This choice of using multiple smaller convolution kernels, rather than a single larger one, optimizes network parameters and enhances classification efficiency.

The feature extraction module consists of five sub-modules, each one is composed of a 1D Resblock, two 1D pooling layers, and a concatenation layer. Within each of these sub-modules, the kernel sizes for the three convolutional layers in the 1D Resblock are set to 32, 11, and 3, with filter numbers of 64, 128, 192, 256, and 320, respectively.

To make predictions, the model utilizes a fully connected layer that integrates information from the preceding layers to produce the final output. This involves a flatten layer, which takes the output from the preceding layer and reshapes it into a single vector, serving as input for the subsequent dense layer.

The initial dense layer comprises 32 nodes and employs Relu activation functions. Following this, a dropout layer with a rate of 0.35 is introduced to mitigate overfitting concerns. Finally, the output is classified into one of the five arrhythmia categories using the SoftMax activation function within the second dense layer.

5 Experiments and Results

5.1 Evaluation Metrics

In this paper, we assessed the performance of our CNN model using several metrics, including overall accuracy (Acc), recall (Sen), and precision (Ppr). These metrics are defined as follows:

$$Acc = \frac{T_p + T_n}{T_p + F_p + F_n + T_n} \tag{3}$$

$$Sen = \frac{T_p}{T_p + F_n} \tag{4}$$

$$Ppr = \frac{T_p}{T_p + F_p} \tag{5}$$

where T_p is true positive, T_n is true negative, F_n is false negative, and F_p is false positive. Indeed, when Precision (Ppr) increases, Sensitivity (Sen) often decreases, and vice versa. The F1 score is a metric that combines Precision and Sensitivity in the following manner:

$$F1 = \frac{2}{\frac{1}{Sen} + \frac{1}{Ppr}} \tag{6}$$

5.2 Training Process

During the training process, a total of 120 training epochs are executed, with the initial raw dataset partitioned into two distinct sets: the training dataset and the testing dataset. Each training epoch employs a batch size of 60. For weight optimization, the Nadam optimization function is utilized, with an initial learning rate set to 0.0001. To improve training efficiency and model performance, an adaptive learning rate mechanism is integrated into the model, dynamically adjusting the learning rate throughout training. Algorithm 1 outlines the evolution of the learning rate over the 120 training epochs. Initially set at 0.0001, the learning rate gradually decreases over epochs, converging to approximately 1×10^{-6} by the final epoch. To evaluate the model's performance and guide training, the cross-entropy loss function, tailored for multiclassification tasks, is chosen as the loss function. The training environment for the model is equipped with a single NVIDIA V100 GPU with 32 GB of RAM.

Algorithm 1 Adaptative_Lr$(current\_epoch, n\_epochs = 120, lr\_start = 1e-4, lr\_end = 1e-6)$

1: $middle = n\_epochs/2$
2: $s = lambda\ x\ : 1/(2 + np.exp(1 - x))$
3: $lr = s(13 * (-current\_epoch + middle)/n\_epochs) * np.abs(lr\_start - lr\_end) + lr\_end$
4: **return** lr

5.3 Results and Comparison

The performance of the proposed method was evaluated using both inter-patient and intra-patient evaluation paradigms. The experimental dataset employed in this paper was sourced from the MIT-BIH dataset, a renowned and extensively used resource in ECG research, known for its precise and comprehensive expert annotations.

Table 3 Evaluation of ECG test dataset performance across varied training-testing ratios in %

Class	Metrics	2 fold	70-30	5 fold	10 fold
N	Accuracy (%)	99,40	99,53	99,56	99,62
	Precision (%)	99,67	99,70	99,72	99,75
	Sensitivity (%)	99,72	99,77	99,66	99,83
	F1-score (%)	99,69	99,74	99,73	99,79
SVEB	Accuracy (%)	99,69	99,74	99,71	99,78
	Precision (%)	95,11	96,77	95,77	97,06
	Sensitivity (%)	93,67	93,53	93,71	94,96
	F1-score (%)	94,39	95,12	94,73	96,00
VEB	Accuracy (%)	99,73	99,80	99,81	99,83
	Precision (%)	97,90	98,34	98,44	99,14
	Sensitivity (%)	98,23	98,81	98,86	98,43
	F1-score (%)	98,06	98,58	98,65	98,78
F	Accuracy (%)	99,80	99,82	99,87	99,85
	Precision (%)	87,63	89,12	92,45	88,24
	Sensitivity (%)	86,53	88,38	91,30	93,75
	F1-score (%)	87,08	88,75	91,87	90,91
Q	Accuracy (%)	99,98	99,98	99,99	99,98
	Precision (%)	00,00	00,00	00,00	00,00
	Sensitivity (%)	00,00	00,00	00,00	00,00
	F1-score (%)	00,00	00,00	00,00	00,00
	Overall Accuracy (%)	99,33	99,43	99,46	99,53

Intra-Patient Classification

To assess the performance of our model under the intra-patient paradigm, we employed k-fold cross-validation with varying values of k, specifically 2, 5, and 10.

Table 3 presents the classification performance of our model across various fold test datasets. It encompasses detailed metrics, including precision, sensitivity, F-score, and overall accuracy.

Our model achieved its highest testing accuracy of 99.53% in the context of five-class classification. The outstanding result was attained by employing a 90:10 training-to-test ratio through 10-fold cross-validation, along with the utilization of an adaptive learning rate function.

Table 4 provides a summary of previous studies focused on the intra-patient paradigm, all of which utilized the MIT-BIH dataset for five-class ECG beat classification. Notably, our model outperformed other deep learning and machine learning architectures, attaining the highest accuracy. Specifically, when compared to CNN models such as [6, 25–28], our model demonstrated superior accuracy, surpassing them by an estimated 0.4%. Moreover, [5] developed a model using a 1D CNN to predict five classes of arrhythmias with an accuracy of 99.48%. However, our model achieved higher accuracy, surpassing it by an estimated 0.05%. Additionally, when

Table 4 Comparative evaluation of our proposed method with prior works in intra-patient paradigm

Method	Clatssifier	Strategy	Acc (%)	Ppr (%)	Sen (%)
Chen et al. [14]	SVM	50-50	98,46	–	98,46
Panday et al. [25]	Deep CNN model	50-50	97.84	87.19	92.99
Our	1D CNN	50-50	99,33	–	–
Panday et al. [25]	Deep CNN model	70-30	98,30	86,06	95,51
Wang et al. [26]	CNN	70-30	99,06	–	–
Our	1D CNN	70-30	99,43	–	–
Kachuee et al. [27]	Deep residual CNN	80-20	93,42	94,30	93,42
Panday et al. [25]	Deep CNN model	80-20	97.64	83.13	95.81
Pham et al. [28]	1D CNN	80-20	98,50	–	–
Our	1D CNN	80-20	99,46	–	–
Elhaj et al. [12]	SVM,NN	90-10	98,91	–	–
Acharya et al. [6]	Deep CNN model	90-10	94,03	97,86	96,71
Panday et al. [25]	Deep CNN model	90-10	97.84	87.19	92.99
Sellami et al. [5]	CNN	90-10	99,48	98.83	96.97
Xu et al. [29]	CNN+BiLSTM	90-10	95,90	96,34	95,90
Our	1D CNN	90-10	99,53	–	–

comparing our model to hybrid models that combine two architectures, CNN and BiLSTM, like [29], our model outperformed them by a substantial margin of at least 3.6% in terms of accuracy.

Inter-Patient Classification

This section focuses on validating our method using the inter-patient paradigm for heartbeat classification. Our proposed model attained an average accuracy of 97.87% for three-class classification. The sensitivity for detecting (SVEB and VEB) reached 75.27% and 90.81%, respectively. Remarkably, the VEB class exhibited superior performance compared to the SVEB class in overall classification accuracy. This discrepancy in performance can be attributed, at least in part, to the smaller sample size but greater subclass diversity of the SVEB class compared to the VEB class.

Our model, as presented in Table 5, which includes five classes, exhibits the lowest average scores, due to its subpar performance on the classes F and Q.

Table 5 presents a comparative analysis of performance between our proposed method and various techniques, utilizing the MIT-BIH arrhythmia dataset and adopting the inter-patient paradigm. Our model with three classes achieved an impressive classification accuracy and demonstrated superior sensitivity for identifying "SVEB" and "VEB" minority classes compared to [14, 30, 31]. Additionally, when considering five classes, our model achieves higher accuracy and precision for "VEB" than all methods with five classes listed in Table 5.

Table 5 Comparative evaluation of our proposed method with prior works in inter-patient paradigm

Method	Class	Classifier	Acc (%)	N		SVEB		VEB		F	
				Ppr (%)	Sen (%)	Ppr (%)	Sen (%)	Ppr (%)	Sen (%)	Ppr (%)	Sen (%)
Chen et al. [14]	3	SVM	93.14	95.42	98.42	38.40	29.50	85.25	70.85	–	–
Zhang et al. [21]	3	Conv1D+ADNN	94.70	98.00	96.20	**90.80**	**78.80**	94.30	**92.50**	–	–
Garcia et al. [30]	3	SVM	92.4	98.00	94.00	53.00	62.00	59.40	87.30	–	–
Lin and Yang [31]	3	LDC	93	**99.30**	91.60	31.60	81.40	73.70	86.20	–	–
Our	3	CNN	**97.87**	98.45	**99.32**	84.47	75.27	**96.53**	90.81	–	–
Li et al. [32]	5	Random Forest, SVM	94.61	**99.73**	94.67	0.16	20.00	**89.78**	94.20	0.52	50.00
Raj and Ray [33]	5	ABC+LSTSVM	96.08	88.50	**98.54**	72.29	52.06	81.59	62.35	17.78	02.31
Sellami et al. [5]	5	CNN	88.34	98.81	88.52	30.44	82.04	72.22	92.05	**26.58**	**68.30**
Cai et al. [34]	5	FFNN+CNN	94.20	98.30	95.80	12.30	33.30	95.60	86.20	0.08	17.60
Zubair et al. [35]	5	CNN	96.36	98.54	91.95	**78.12**	**86.66**	41.74	77.75	20.58	68.04
Our	5	CNN	**96.51**	98.05	98.29	76.08	79.80	88.21	**93.39**	00.00	00.00

6 Conclusions and Future Work

In this paper, we have proposed a 1-D Convolutional Neural Network (CNN) for the automatic classification of ECG signals. Our goal was to categorize five distinct types of arrhythmias. To enhance the model's efficacy, we employed a range of preprocessing techniques on the ECG signals, including denoising, segmentation, and data balancing. We conducted our experiments using the MIT-BIH Arrhythmia dataset to validate the effectiveness of our method. The experimental findings underscore the remarkable efficacy of our model, with an impressive overall accuracy of 99.53% for intra-patient 10-fold and 97.87% for inter-patient paradigm.

It's noteworthy that the intra-patient evaluation outperformed the inter-patient evaluation. However, we raised concerns about the potential overlap between the testing set and the training set in the intra-patient paradigm, which could potentially bias and inflate the results. Therefore, the inter-patient paradigm may be more reliable for unbiased evaluation.

Despite these concerns, our proposed method continues to surpass the performance of previous methods. Furthermore, our ECG arrhythmia classifier exhibits potential for diverse biomedical applications, including sleep staging and assisting medical experts in detecting cardiac arrhythmia more accurately, possibly through the use of medical robots that monitor ECG signals.

In our future work, we intend to introduce an alternative architecture for inter-patient ECG classification in arrhythmia detection, using a 1-D Convolutional Neural Network (CNN). This will involve harnessing the morphological and temporal characteristics derived from RR intervals to capture more information and improve the accuracy of arrhythmia identification.

References

1. World Health Organization: Cardiovascular Diseases World Health Organization, Switzerland (2021)
2. ANSI-AAMI: Testing and reporting performance results of cardiac rhythm and ST segment measurement algorithms. American National Standards Institute, Inc. (ANSI), Association for the Advancement of Medical Instrumentation (AAMI), ANSI/AAMI/ISO (1998–2008)
3. De Chazal, P., O'Dwyer, M., Reilly, R.B.: Automatic classification of heartbeats using ECG morphology and heartbeat interval features. IEEE Trans. Biomed. Eng. **51**, 1196–1206 (2004)
4. Pradhan, G., Singh, P., Shahnawazuddin, S.: An efficient ECG denoising technique based on non-local means estimation and modified empirical mode decomposition. Circuits Syst. Signal Process. (2018)
5. Sellami, A., Hwang, H.: A robust deep convolutional neural network with batch-weighted loss for heartbeat classification. Expert Syst. Appl. **122**, 75–84 (2019)
6. Acharya, U.R., Oh, S.L., Hagiwara, Y., Tan, J.H., Adam, M., Gertych, A., Tan, R.S.: A deep convolutional neural network model to classify heartbeats. Comput. Biol. Med. **89**, 389–396 (2017)
7. Dong, Q., Gong, S., Zhu, X.: Imbalanced deep learning by minority class incremental rectification. IEEE Trans. Pattern Anal. Mach. Intell. **41**, 1367–1381 (2018)
8. Chang, S., Li, Y., Shen, J.S., Feng, J., Zhou, Z.: Contrastive attention for video anomaly detection. IEEE Trans. Multimed., 1–10 (2021)
9. Luz, E.J.S., Schwartz, W.R., Cámara-Chávez, G., Menotti, D.: ECG-based heartbeat classification for arrhythmia detection: a survey. Comput. Methods Programs Biomed. **127**, 144–164 (2016)
10. Nguyen, T.Q., Luo, S., Afonso, V.X., Tompkins, W.J.: ECG beat detection using filter banks. IEEE Trans. Biomed. Eng. (1999)
11. Pan, J., Tompkins, W.J.: A real-time QRS detection algorithm. IEEE Trans. Biomed. Eng. BME-32, 230–236 (1985)
12. Elhaj, F.A., Salim, N., Harris, A.R., Swee, T.T., Ahmed, T.: Arrhythmia recognition and classification using combined linear and nonlinear features of ECG signals. Comput. Methods Programs Biomed. **127**, 52–63 (2016)
13. Shi, H., Wang, H., Huang, Y., Zhao, L., Qin, C., Liu, C.: A hierarchical method based on weighted extreme gradient boosting in ECG heartbeat classification. Comput. Methods Programs Biomed. **171**, 1–10 (2019)
14. Chen, S., et al.: Heartbeat classification using projected and dynamic features of ECG signal. Biomed. Signal Process. Control **31**, 165–173 (2017)
15. Dang, H., Sun, M., Zhang, G., Zhou, X., Chang, Q., Xu, X.: A novel deep convolutional neural network for arrhythmia classification. In: Proceedings of the 2019 International Conference on Advanced Mechatronic Systems (ICAMechS), Shiga, Japan, 26–28 August 2019, pp. 7–32. IEEE, Piscataway, NJ, USA (2019)
16. Oh, S.L., Ng, E.Y., San Tan, R., Acharya, U.R.: Automated beat-wise arrhythmia diagnosis using modified U-net on extended electrocardiographic recordings with heterogeneous arrhythmia types. Comput. Biol. Med. **105**, 92–101 (2019)
17. Übeyli, E.D.: Combining recurrent neural networks with eigenvector methods for classification of ECG beats. Digit. Signal Process. **19**, 320–329 (2009)
18. Yildirim, Ö., Baloglu, U.B., Tan, R.S., Ciaccio, E.J., Acharya, U.R.: A new approach for arrhythmia classification using deep coded features and LSTM networks. Comput. Methods Programs Biomed. **176**, 121–133 (2019)
19. Yildirim, Ö.: A novel wavelet sequence based on deep bidirectional LSTM network model for ECG signal classification. Comput. Biol. Med. **96**, 189–202 (2018)
20. Oh, S.L., Ng, E.Y., San Tan, R., Acharya, U.R.: Automated diagnosis of arrhythmia using combination of CNN and LSTM techniques with variable length heart beats. Comput. Biol. Med. **102**, 278–287 (2018)

21. Zhang, J., Liu, A., Liang, D., Chen, X., Gao, M.: Interpatient ECG heartbeat classification with an adversarial convolutional neural network. J. Healthc. Eng. **2021**, 9946596 (2021)
22. Wang, T., Lu, C., Sun, Y., Yang, M., Liu, C., Ou, C.: Automatic ECG classification using continuous wavelet transform and convolutional neural network. Entropy **23**, 119 (2021)
23. Goldberger, A.L., Amaral, L.A., Glass, L., Hausdorff, J.M., Ivanov, P.C., Mark, R.G., Mietus, J.E., Moody, G.B., Peng, C.K., Stanley, H.E.: PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. Circulation **101**, e215–e220 (2000)
24. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: synthetic minority oversampling technique. J. Artif. Intell. Res. **16**, 321–357 (2002)
25. Pandey, S.K., Janghel, R.R.: Automatic detection of arrhythmia from imbalanced ECG database using CNN model with SMOTE. Australas. Phys. Eng. Sci. Med. **42**, 1129–1139 (2019)
26. Wang, H., Shi, H., Chen, X., Zhao, L., Huang, Y., Liu, C.: An improved convolutional neural network-based approach for automated heartbeat classification. J. Med. Syst. **44**(2), 35 (2019)
27. Kachuee, M., Fazeli, S., Sarrafzadeh, M.: ECG heartbeat classification: a deep transferable representation. In: 2018 IEEE International Conference on Healthcare Informatics (ICHI), pp. 443–444. IEEE (2018)
28. Pham, B.-T., Le, P.T., Tai, T.-C., Hsu, Y.-C., Li, Y.-H., Wang, J.-C.: Electrocardiogram heartbeat classification for arrhythmias and myocardial infarction. Sensors **23**, 2993 (2023)
29. Xu, X., Jeong, S., Li, J.: Interpretation of electrocardiogram (ECG) rhythm by combined CNN and BiLSTM. IEEE Access **8**, 125380–125388 (2020)
30. Garcia, G., Moreira, G., Menotti, D., Luz, E.: Inter-patient ECG heartbeat classification with temporal VCG optimized by PSO. Sci. Rep. **7**, 1–11 (2017)
31. Lin, C.-C., Yang, C.-M.: Heartbeat classification using normalized RR intervals and morphological features. Math. Probl. Eng. **2014** (2014)
32. Li, T., Zhou, M.: ECG classification using wavelet packet entropy and random forests. Entropy **18**, 285 (2016)
33. Raj, S., Ray, K.C.: A personalized arrhythmia monitoring platform. Sci. Rep. **8**, 11395 (2018)
34. Cai, J., Zhou, G., Dong, M., Hu, X., Liu, G., Ni, W.: Real-time arrhythmia classification algorithm using time-domain ECG feature based on FFNN and CNN. Math. Probl. Eng. **2021**, 6648432 (2021)
35. Zubair, M., Yoon, C.: Cost-sensitive learning for anomaly detection in imbalanced ECG data using convolutional neural networks. Sensors **22**, 4075 (2022)

Proposed Hybrid Model of Focused Crawler Based on Images Containing Tables

Hayat Ouadi, Ilhame El Farissi, and Ilham Slimani

Abstract The increasing amount of online data has led to a greater demand for web crawlers that can effectively extract information from web pages. One common challenge is dealing with images that contain tables, as traditional text-based crawlers struggle to process them. To address this issue, we have created a specialized hybrid crawler specifically designed to target images with tables. This advanced crawler utilizes sophisticated image processing techniques for accurate data extraction. By combining content-based image retrieval and machine learning algorithms, our approach enables the crawler to recognize and categorize images based on their visual features. Important testing on financial web pages has demonstrated the remarkable accuracy of our crawler in retrieving relevant images containing tables.

Keywords Focused crawler · Image processing · Content-based image retrieval

1 Introduction

The number of websites on the World Wide Web has reached an impressive 1.14 billion, however, only 17% considered to be active [1]. With such a large amount of information readily available, it is becoming more and more difficult to search for useful information. For this reason, it is important to develop document discovery mechanisms based on intelligent techniques such as focused crawling. In the classical sense, crawling is the act of using automated software agents to gather internet data. In other words, it's a relatively simple automated program, or script, that methodically scans through webpages.

This is where the concept of focused crawling comes into play. Focused crawling is a document discovery method that uses intelligent techniques to gather specific types of data [2] The main difference between classical crawling and focused crawling lies in their approach to data collection. Classical crawling collects all types of data indiscriminately, while focused crawling only collects data that is relevant to a

H. Ouadi (✉) · I. E. Farissi · I. Slimani
SmartICT Laboratory, ENSAO, Mohammed First University, Oujda, Morocco
e-mail: hayat.ouadi@ump.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_13

specific topic or subject. This ensures that time and memory are optimized, as the crawler only exports data that is truly useful for the intended purpose.

The rest of this paper is structured as follows: Sect. 2 reviews types of web crawlers. The classic crawler design is discussed in Sect. 3. The design with the presentation of the new learning crawler proposed in this work, are presented in Sect. 4. A methodology proposed of the crawlers considered in this work is presented in Sect. 5 followed by results in Sect. 6.in the last section, we discussed conclusion and issues for further research.

2 Related Work

In order to optimize the crawling process and retrieve only the most relevant resources, it is important to navigate the website in a more targeted and purposeful manner. This can be achieved by using different crawling strategies, such as focusing on specific sections of the website, following particular links, or using filtering techniques to exclude unwanted pages.

2.1 Classic Focused Crawlers

Classic focused crawlers, also known as classic heuristic focused crawlers, primarily rely on the hyperlink structure and text content of webpages. These crawlers can be tailored to specific HTML tags or URLs within a webpage. One of the key algorithms used by hyperlink focused crawlers is the Page Rank (PR) algorithm, which seeks to determine the most important pages by maximizing the power of paths.

One such focused web crawler that uses the PR algorithm is the work of Ankit and Pawan [3] Their experiments show a marked difference in performance metrics and demonstrate that their proposed algorithm outperforms established algorithms like Page Rank and Weighted Page Rank.

2.2 Semantic Based Focused Crawlers

Semantic-based focused crawlers, also known as conceptual semantic analysis-based crawlers, are considered to be among the most effective focused crawlers. These crawlers determine the relevance of a page to a specific topic based on the similarity of the document. It is important to understand that two documents can have different wording, but still have a similar meaning. This is referred to as semantic similarity. For

example, one document may contain keywords such as "virus," "health," and "hospitalization," while another document may contain "confinement," "mask," and "sterilize." Although these lists of keywords have different wording, they are semantically similar.

The similarity of documents can be calculated using either the Vector Space Model (VSM) or specialized models such as the one proposed by Liu and He [4]. This specialized model has been shown to be more effective than the VSM in determining document similarity. By utilizing semantic-based focused crawlers, it is possible to effectively identify pages that are relevant to a specific topic, even if they have different wording.

2.3 Intelligent Focused Crawlers

Intelligent focused crawler or learned crawler is a crawler supervised by a training set to specify user's preferences by a set of selected documents. we have for instance, the work presented by Kaleel and Sheen [2], it outlines the development of an intelligent focused crawler employing Reinforcement Learning with a decaying ϵ-greedy policy, specifically designed to enhance the performance of vertical search engines. The study demonstrates improved efficiency in data gathering and user-specified preference identification compared to traditional ϵ-greedy approaches.

To sum up, in our study, we used a hybrid focused web crawler that combines intelligent focused crawler and classic focused crawler with Specify only URL of images.

3 Classic Crawler Design

The following are the key elements of crawler design [5]:

a. **Input**: The input to a crawler consists of a set of starting URLs, known as seed URLs, and, in the case of focused crawlers, a topic description. This description can take the form of a list of keywords for classic and semantic focused crawlers or a training set for machine learning-based crawlers.
b. **Page Downloading**: The links within the downloaded pages are extracted and placed in a queue. For non-focused crawlers, these links are processed in a first-in, first-out manner. In the case of focused crawlers, the queue entries are reordered based on criteria such as content relevance or importance. The focused crawler may also decide to exclude certain links from further expansion. Similarly, a generic crawler may also apply importance criteria to determine which pages are worth crawling and indexing.

c. **Priority Assignment**: The extracted URLs from the downloaded pages are placed in a priority queue, and their priorities are determined based on the type of crawler and user preferences. These priorities range from simple criteria, such as page importance or relevance to the query topic, computed by matching the query with the page or anchor text, to more complex criteria determined through a learning process.
d. **Expansion**: URLs are selected for further expansion, and steps (b) to (d) are repeated until the desired number of pages have been downloaded or system resources are exhausted, whichever comes first.

The algorithm depicted in Fig. 1 delineates the traditional steps involved, including the initiation from seed URLs, the recursive exploration of hyperlinks, and the strategic prioritization of pages for crawling. This classic approach is foundational to information retrieval, enabling efficient indexing and ensuring the comprehensive coverage of the vast expanse of the World Wide Web.

The following steps outline the basic algorithm for a web crawler:

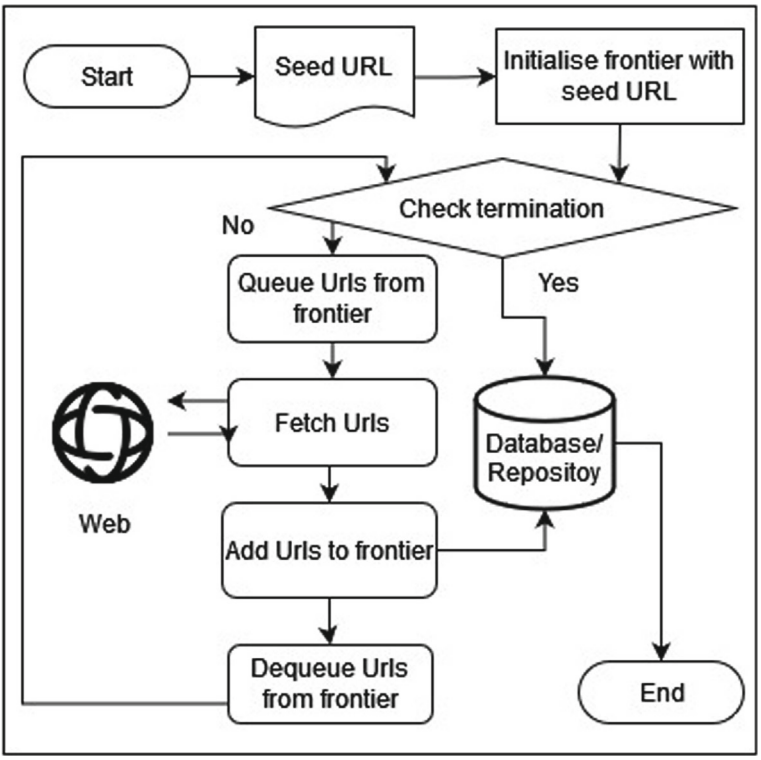


Fig. 1 Classic web crawler design [5]

1. **Start with a seed set of URLs**: The crawler is given a starting set of URLs known as seed URLs. These URLs serve as the starting point for the crawler to begin its exploration.
2. **Extract links from the seed pages**: The crawler downloads the content of each seed URL and extracts all the links from the page. These links are then added to a queue for further exploration.
3. **Download and process the content of each page**: The crawler visits each URL in the queue, downloads the content of the page, and processes the information to extract relevant data. This may involve parsing the HTML structure of the page, extracting text and images, and processing any other type of media.
4. **Prioritize pages to visit next**: The crawler determines the priority of the pages in the queue, deciding which pages to visit next. This can be based on factors such as the relevance of the page to the search query, the freshness of the content, or the importance of the page as determined by the algorithm.
5. **Repeat the process**: The crawler repeats the process of extracting links, downloading content, and prioritizing pages until it has covered a sufficient amount of the web or until it reaches a stopping criteria, such as a limit on the number of pages visited.

The algorithm underlying web crawling forms the backbone of the entire process. However, tweaks and modifications can be implemented to enhance the speed and accuracy of the crawler.

4 Focused Crawler Algorithm

The basic algorithm for a focused web crawler is similar to that of a general web crawler, with some additional steps or modifications to ensure that the crawler focuses on the desired content:

1. **Start with a seed set of URLs**: The focused crawler is given a set of seed URLs that are relevant to the topic or information being targeted.
2. **Extract links from the seed pages**: The focused crawler downloads the content of each seed URL and extracts all the links from the page. These links are then added to a queue for further exploration.
3. **Filter and prioritize the links**: The focused crawler uses various algorithms and heuristics to filter and prioritize the extracted links based on their relevance to the target topic or information. For example, the crawler may use keyword matching, semantic analysis, or machine learning techniques to determine the relevance of each link.
4. **Download and process the content of each page**: The focused crawler visits each URL in the queue, downloads the content of the page, and processes the information to extract relevant data. This may involve parsing the HTML structure of the page, extracting text and images, and processing any other type of media.

5. **Evaluate and refine the relevance of each page**: The focused crawler uses various techniques to evaluate the relevance of each page to the target topic or information. Based on the evaluation, the crawler may refine its approach, adjusting the algorithms and heuristics used to filter and prioritize links.
6. **Repeat the process**: The focused crawler repeats the process of extracting links, filtering and prioritizing links, downloading content, and evaluating and refining relevance until it has covered a sufficient amount of the web or until it reaches a stopping criteria, such as a limit on the number of pages visited.

Figure 2 shows the architecture of our focused crawler [6].

The algorithm for a focused web crawler is designed to target specific types of content or information, resulting in a more efficient and effective exploration of the web. By prioritizing and focusing on relevant content, a focused web crawler can produce a higher quality dataset with less noise and less redundancy.

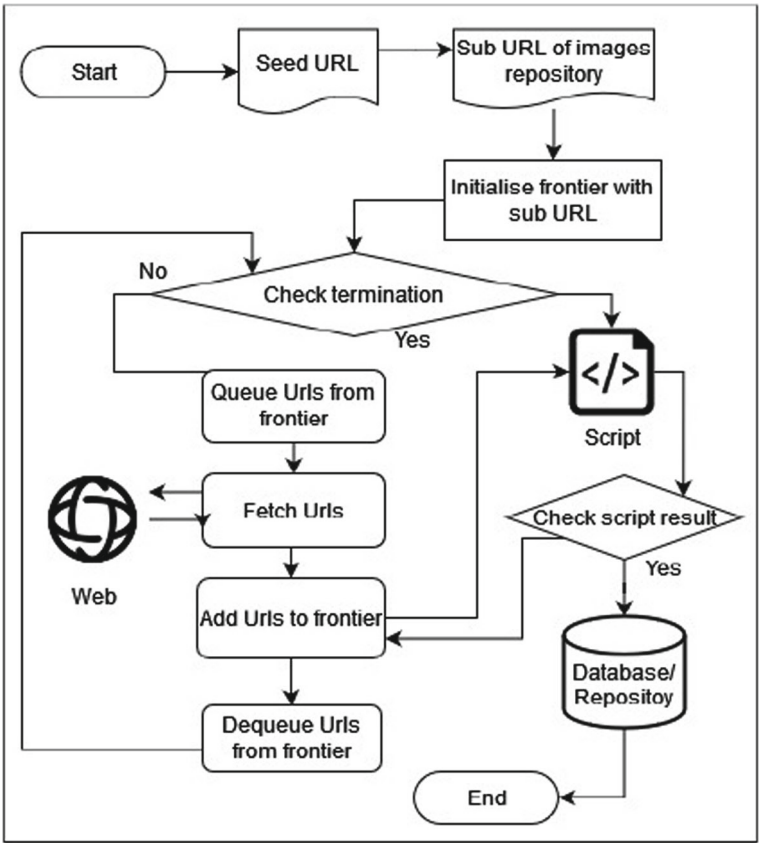


Fig. 2 Proposed Hybrid model of Focused crawler based on images containing tables

5 Methodology

We implemented a Hybrid focused web crawler to download images containing tables from a list of finance-related websites. The crawler was written in Java 8 using the Jsoup web scraping library and the OpenCV image processing library.

5.1 Crawling

We are using **Jsoup** in Java to crawl images following these steps:

a. Import Jsoup library in your Java project. We are downloading the latest version of Jsoup from the official website.
b. Connect to the web page we want to crawl using Jsoup's connect () method. This method returns a Document object that represents the parsed HTML content of the page.

```
Document doc = Jsoup.connect ("https://www.example.com").get();
```

c. Use Jsoup's selector syntax to find all the image elements on the page. For example, to find all the img elements, we used the select() method with the CSS selector img.

```
Elements images = doc.select("img");
```

d. Loop through the images collection and extract the URLs of the image files. We use the attr() method to get the value of a specific attribute of an element. In this point, we are sending our image URL to the classifier that return TRUE/FALSE according to the existence of the table in the image.

```
For (Element image: images) {
String imageUrl = image.attr("src");
isTableExist(imageUrl);
}
So, if this method returns true we can have the next step.
```

e. To download the images, we used Java's built-in URL and URLConnection classes.

5.2 Tables Detection

To detect images containing tables, we used OpenCV's Tesseract to perform contour detection on each downloaded image. We applied a threshold to each image to convert it to grayscale and remove any background noise, and then used the **detectMultiScale()** function to detect any shapes with four sides or more. We then applied additional filters to remove any false positives, such as shapes with a small area or aspect ratio, and those with overlapping or nested contours (see Fig. 3).

The method we used is based on OpenCV's Haar Cascade Classifier [7] to detect regions in the image that may contain tables. The Haar Cascade Classifier is a machine learning-based algorithm used for object detection. It works by identifying specific features within an image that are characteristic of the object being searched for. In this case, we use the classifier to recognize features of tables, such as horizontal and vertical lines.

1. Load the image:

```
image = cv2.imread("path/to/image.jpg")
```

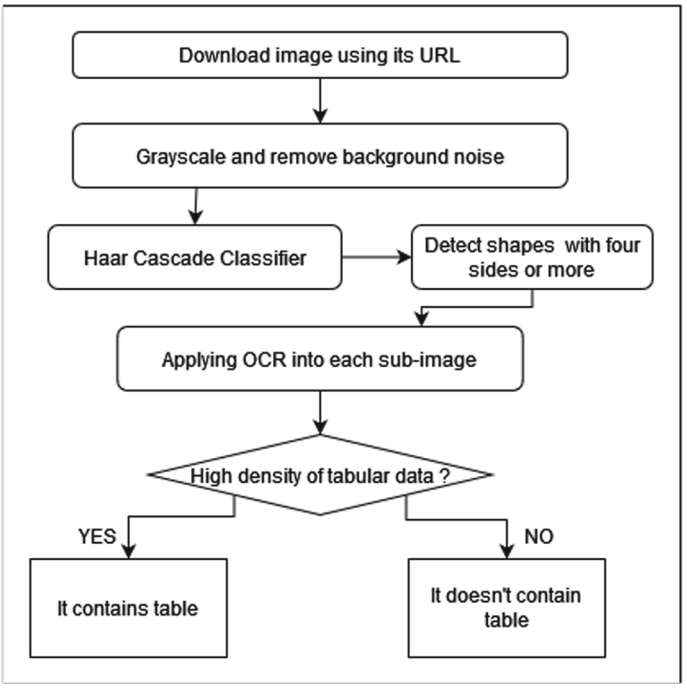


Fig. 3 Algorithm to detect images containing tables

2. Convert the image to grayscale:

```
gray_image = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
```

3. Load the Haar Cascade classifier for tables:

```
cascade_file = "path/to/haarcascade_table.xml"
```

```
table_cascade = cv2.CascadeClassifier(cascade_file)
```

4. Detect tables in the image:

```
tables = table_cascade.detectMultiScale(gray_image, scaleFactor = 1.1,
minNeighbors = 5)
```

Then, it extracts the sub-images of those regions and applies Tesseract OCR to recognize the text in the images. If the text contains a high density of tabular data, it may indicate that the region contains a table.

6 Results

In this section, we presents the results and performance of the proposed method. The experiments and implementation were based on the Tesseract and Jsoup. We crawled a total of 27 finance-related websites and downloaded a total of 5163 images. From these images, we were able to detect and extract 5986 tables containing financial data. We have some images with more than one Table 1.

We crawled a total of 5163 images, of which 1751 images were not related to our goal. These images may include logos and marketing images.

Table 1 Number of gathered images in each step

Step number	Step description	Gathered images
1	Crawling all images	5163
2	Filtering images containing tables	3412

The number of relevant results is related to the choice of pages to crawl. We only select pages that contain tables and are allowed to be crawled in their **robots.txt** file.

The effects of proposed services on the focused crawling performance can be measured by the harvest rate (HR). HR represents the percentage of relevant retrieved images (n_{rImg}) in comparison with all retrieved images (N) during the crawling as shown in Eq. (1) [8].

$$HR = \frac{\text{Number of Images with Detected Tables}}{\text{Total Number of Crawled Images}} * 100 \quad (1)$$

In Our proposed service, the harvest rate is 66%, which means that we are filtering more than 30% of crawled images.

Our results demonstrate the feasibility of using an image web crawler to extract financial data from images containing tables. By implementing a combination of image detection and data extraction algorithms, we were able to detect and extract a large number of tables with high accuracy.

7 Implications for Financial Research

Our approach has several implications for financial research. Firstly, it allows for the extraction of financial data from sources that are not currently included in existing financial databases. This could potentially lead to the discovery of new financial patterns or anomalies that were previously undiscovered. Secondly, it allows for the analysis of financial data in a more granular and fine-grained way, by extracting data from specific tables and contexts rather than relying on pre-defined metrics or summaries. Finally, it could potentially enable the development of new financial data products or services that leverage the extracted data in innovative ways.

8 Conclusion

Despite the promising results, our approach has several limitations that should be addressed in future work. Firstly, our sample size was limited to a small set of finance-related websites in French language, and the results may not generalize to other types of websites or images. Secondly, the accuracy of the data extraction may depend on the quality of the original images, and may not be as accurate for low-resolution or blurry images.

In future work, we plan to address these limitations by exploring alternative image processing techniques and expanding the set of crawled websites.

In summary, a focused web crawler that relies on image recognition to extract financial data from tables can be a valuable tool for web-based data collection. These crawlers use advanced technologies such as machine learning and computer vision

to extract financial information from tables embedded in images, which traditional text-based crawlers cannot easily access.

Using such crawlers can significantly enhance the accuracy and efficiency of financial data collection for analytical purposes. However, the performance of the crawler will depend on the quality of the image and the complexity of the table structure. Thus, it is crucial to continuously refine and improve the crawler algorithm to ensure the best performance.

Overall, employing a focused web crawler based on image recognition to collect financial data has the potential to revolutionize financial data collection and analysis, providing valuable insights to support investment decisions and business strategies.

References

1. NJ: How Many Websites Are There in the World? (2023). Siteefy. https://siteefy.com/how-many-websites-are-there/. Accessed 15 Oct 2023
2. Kaleel, P.B., Sheen, S.: Focused crawler based on reinforcement learning and decaying epsilon-greedy exploration policy. Int. Arab J. Inf. Technol. **20**(5), 819–830 (2023). https://doi.org/10.34028/iajit/20/5/14
3. Vidyarthi, A., Singh, P.: Chapter fourteen—power rank: an interactive web page ranking algorithm. In: Patgiri, R., Deka, G.C., Biswas, A. (eds.) Advances in Computers, vol. 128. Principles of Big Graph: In-depth Insight, vol. 128, pp. 353–379. Elsevier (2023). https://doi.org/10.1016/bs.adcom.2021.10.008.
4. Liu, W., et al.: A focused crawler based on semantic disambiguation vector space model. Complex Intell. Syst. **9**(1), 345–366 (2023). https://doi.org/10.1007/s40747-022-00707-8
5. Kim, K.S., Kim, K.Y., Lee, K.H., Kim, T.K., Cho, W.S.: Design and implementation of web crawler based on dynamic web collection cycle. In: Presented at the International Conference on Information Networking, pp. 562–566 (2012). https://doi.org/10.1109/ICOIN.2012.6164440
6. Lu, H., Zhan, D., Zhou, L., He, D.: An improved focused crawler: using web page classification and link priority evaluation. Math. Probl. Eng. **2016** (2016). https://doi.org/10.1155/2016/6406901
7. Dhandapani, R., Humaid Al-Ghafri, S.M.: Implementation of facial mask detection and verification of vaccination certificate using Jetson Xavier kit. In: Presented at the IOP Conference Series: Earth and Environmental Science (2022). https://doi.org/10.1088/1755-1315/1055/1/012013
8. Du, Y., Liu, W., Lv, X., Peng, G.: An improved focused crawler based on Semantic Similarity Vector Space Model. Appl. Soft Comput. **36**, 392–407 (2015). https://doi.org/10.1016/j.asoc.2015.07.026

Vehicle Detection in Stereoscopic Images Using Symmetry-Based Approach

El Asri Soufiane and Zebbara Khalid

Abstract This study introduces a unique symmetry-based approach for vehicle detection in stereoscopic images, a critical advancement for driver assistance systems. Distinct from existing techniques, this method innovatively combines the Canny operator with corner points to create a more precise and computationally efficient vehicle detection system. The method uses the canny operator and corner points to identify contours and then employs symmetry maps and moment calculations to link these contour points. The method was tested using the KITTI stereo dataset, which contains real-world driving scenarios, and compared with established methods like the Harris corner detector and Scale-Invariant Feature Transform (SIFT). The new approach showed superior accuracy, with an average precision (AP) of 92.3% and average recall (AR) of 87.5%, outperforming the Harris detector (AP = 85.7%, AR = 81.2%) and SIFT (AP = 89.1%, AR = 84.6%). Additionally, it was more computationally efficient, processing frames in 0.034 s on average, faster than both the Harris detector (0.057 s) and SIFT (0.049 s). Further validation was done using the Middlebury Stereo Dataset and the Karlsruhe Urban Stereo Dataset. The method continued to demonstrate high performance, achieving AP values of 87.2% and 84.6% on these datasets, respectively, again outperforming the Harris detector and SIFT. This indicates the method's robustness and effectiveness in various scenarios.

Keywords Autonomous driving · Road safety · Obstacle detection · Intelligent vehicle · Stereoscopic image · Image processing · Symmetry detection · Canny operator · Vehicle recognition

E. A. Soufiane (✉) · Z. Khalid
Faculty of Sciences, Agadir, Morocco
e-mail: soufiane.elasri@edu.uiz.ac.ma

Z. Khalid
e-mail: k.zebbara@uiz.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_14

1 Introduction

Advancements in driver assistance systems are pivotal for road safety, yet the reliable detection of vehicles in stereoscopic images remains a significant technical challenge. This challenge is compounded by the complexity of real-world environments where vehicles must be differentiated from dynamically changing backgrounds.

Addressing this, our study introduces a pioneering approach that utilizes a unique combination of the Canny operator [4] and corner points, integrated with symmetry maps for enhanced detection accuracy. This novel methodology stands out in its ability to more precisely identify vehicles in diverse scenarios, setting a new benchmark for computational efficiency in stereoscopic image processing.

The implications of this research extend beyond technical innovation, offering tangible benefits to driver assistance systems, potentially leading to more advanced safety mechanisms in the evolving landscape of autonomous vehicles.

The paper is structured as follows: Sect. 2 provides a literature review, Sect. 3 describes our methodology, Sect. 4 presents the results, Sect. 5 discusses these findings, and Sect. 6 concludes the paper with future research directions.

Recent advancements in AI and machine learning for vehicle detection, such as the novel neural network training techniques and pruning algorithms explored by Movassagh et al. [10] and Alzubi et al. [2], provide a relevant backdrop to our study.

2 Literature Review

In our exploration of vehicle detection in stereoscopic images, this article introduces a unique approach grounded in symmetry principles. Our methodology employs the Canny operator and corner points to discern contours within images, establishing connections between contour points through the utilization of symmetry maps and moment calculations.

Symmetry-based methodologies have exhibited notable success in enhancing object detection [5] accuracy across diverse applications, including face recognition and object tracking. By capitalizing on the inherent symmetry of objects, our approach demonstrates improved resilience to complex backgrounds and lighting variations, factors that often challenge vehicle detection in stereoscopic images.

The realm of real-time object detection has extensively embraced Convolutional Neural Networks (CNNs), while deep neural networks have advanced spatial understanding through depth estimation techniques. Recurrent Neural Networks (RNNs) and Generative Adversarial Networks (GANs) have proven beneficial for trajectory planning and control. Reinforcement learning algorithms, such as Deep Q-Networks (DQN) and Proximal Policy Optimization (PPO), have facilitated self-learning obstacle avoidance behaviors. Noteworthy advancements in model training have been achieved through the utilization of large annotated datasets (e.g., KITTI, ApolloScape [6]) and simulators (e.g., CARLA, AirSim), significantly improving the robustness

of the models. However, challenges persist, particularly in adapting models to diverse environments and addressing complexities in road interactions.

Past approaches to vehicle detection in stereoscopic images have encompassed various methods, including stereo matching and feature extraction. Nevertheless, these methodologies are associated with limitations concerning both accuracy and computational efficiency. Our proposed approach addresses these concerns, presenting a more efficient and accurate solution to the intricate task of vehicle detection in stereoscopic images.

In line with contemporary research trends, our study shares similarities with the work of Alzubi et al. [1], who investigate distracted driver detection using energy-efficient convolutional neural networks.

3 Methodology

This section outlines the methodology employed to implement our proposed approach for vehicle detection in stereoscopic images. The methodology encompasses the following steps.

3.1 Image Preprocessing

In the image preprocessing stage, a diverse set of stereoscopic images capturing urban driving scenarios was collected. To rectify distortions and ensure accurate depth estimation, camera calibration was performed, yielding intrinsic and extrinsic parameters. The images were then resized to a standardized 640×480 pixels and converted to grayscale to simplify subsequent processing. Gaussian smoothing was applied to reduce noise, and contrast enhancement techniques were employed to address variations in lighting conditions. Adaptive thresholding optimized edge detection under different illuminations. Optionally, data augmentation techniques, including random rotations, flips, and lighting changes, were applied to introduce variability in the dataset. This comprehensive preprocessing aimed to standardize and enhance the input data, establishing optimal conditions for subsequent computer vision tasks, particularly in the context of vehicle detection in stereoscopic images.

3.2 Contour Detection

Utilizing the Canny edge detector and the Shi-Tomasi corner detection algorithm [11], our methodology extracted significant features and established correspondences between left and right views [8].

The Canny edge detector identified essential edge points through a series of steps, including Gaussian smoothing, gradient calculation, non-maximum suppression, and thresholding. Concurrently, the Shi-Tomasi corner detection algorithm identified key points based on the gradient of intensity changes. The combination of these techniques facilitated the extraction of distinctive feature points, serving as reference points for establishing correspondences between views and enabling subsequent depth estimation and 3D reconstruction (see Fig. 1).

3.3 Contour Association

In our methodology, we employed the stereo block matching algorithm to calculate the disparity between the left and right views of the stereo images. The stereo block matching algorithm compares corresponding blocks of pixels in the left and right views and estimates the disparity, which represents the horizontal shift between the pixel locations in the two views. This disparity information provides valuable depth cues for reconstructing the 3D structure of the scene.

By utilizing the calculated disparity values, we were able to estimate the 3D coordinates of the key points identified in the images. Using the disparity information and the known baseline distance between the stereo camera setup, we could triangulate the key points to determine their respective 3D positions in the scene. This enabled us to create a three-dimensional representation of the objects or scene captured by the stereo images.

Additionally, we employed the Haar wavelet transform to calculate the symmetry map for each contour. The Haar wavelet transform is a technique that analyzes the variation in pixel intensities and identifies regions of symmetry within an image or contour. By computing the symmetry map, we were able to quantify the degree

Fig. 1 Contour detection using the canny detector

of symmetry for each contour, providing useful information for further analysis or object recognition tasks.

By integrating the disparity calculation, 3D coordinate estimation, and symmetry map calculation, we were able to leverage the power of stereo vision and wavelet-based symmetry analysis to obtain detailed depth information and symmetry characteristics of the objects or scene captured by the stereo images. These techniques contributed to a comprehensive understanding of the spatial structure and symmetry properties of the scene (see Fig. 2).

3.4 Object Classification and Tracking

In our approach, we utilized the Scale-Invariant Feature Transform (SIFT) operator for both object classification and tracking tasks.

The SIFT operator is a popular feature detection and description technique that is robust to changes in scale, rotation, and affine transformations. It extracts distinctive local features from an image, known as keypoints or SIFT descriptors, which are invariant to these transformations.

For object classification, we first identified keypoint correspondences between the reference object or objects of interest and the input images. This was done by extracting SIFT keypoints from both the reference object and the image using the SIFT operator. The keypoints were then matched using techniques such as nearest neighbor matching or RANSAC (Random Sample Consensus) to establish reliable correspondences.

Once the keypoint correspondences were established, we employed a classification algorithm, such as Support Vector Machines (SVM) or k-nearest neighbors (k-NN), to classify the matched keypoints. Each keypoint was associated with a class label based on the reference object it corresponded to. This allowed us to classify the objects present in the image based on the matched keypoints and their associated labels.

For object tracking, we employed a similar methodology. We first extracted SIFT keypoints from the initial frame or frames containing the target object. These keypoints served as reference features for tracking. In subsequent frames, we again

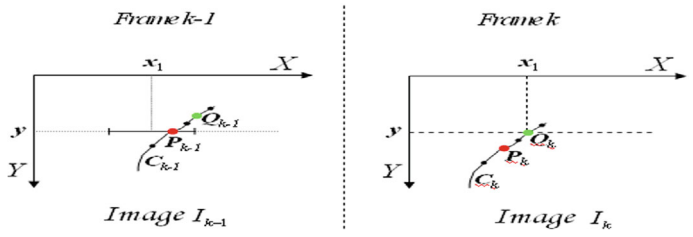


Fig. 2 Association between contour points (Calculation of the association)

applied the SIFT operator to extract keypoints and matched them with the reference keypoints from the initial frame using techniques such as nearest neighbor matching.

The matched keypoints between frames provided the necessary information for tracking the object's position and motion. We could use techniques such as optical flow or Kalman filters to estimate the object's trajectory and update its position over time.

By leveraging the SIFT operator for both object classification and tracking, we obtained robust and distinctive features that were invariant to scale, rotation, and affine transformations. This enabled accurate classification and reliable tracking of objects in various real-world scenarios [9].

3.5 Evaluation

To assess the effectiveness of our proposed approach, we conducted performance evaluations using the KITTI stereo dataset, a widely used benchmark dataset that comprises a large number of stereoscopic images capturing urban driving scenes.

During the evaluation, we employed various metrics, including precision and recall, to quantitatively measure the accuracy and robustness of our approach.

Precision is a metric that measures the ratio of correctly detected vehicles to the total number of detected vehicles. It assesses the ability of our approach to accurately identify and classify vehicles in the images. A higher precision indicates a lower rate of false positives, highlighting the precision of our vehicle detection and classification results.

Recall, on the other hand, evaluates the ratio of correctly detected vehicles to the total number of ground truth vehicles in the dataset. It measures the comprehensiveness and effectiveness of our approach in capturing all the vehicles present in the scene. A higher recall indicates a lower rate of false negatives, indicating the ability of our approach to detect a significant portion of the vehicles in the dataset.

By calculating precision and recall metrics, we were able to quantitatively assess the performance of our approach on the KITTI stereo dataset. These metrics provided valuable insights into the accuracy, completeness, and overall effectiveness of our proposed methodology for vehicle detection and classification in urban driving scenes captured by stereoscopic images.

3.6 Computational Complexity

We implemented our proposed approach in Python using the OpenCV [3] and NumPy libraries. The computational complexity of our approach was optimized using parallel processing techniques.

4 Results

Our evaluation employed a comprehensive analysis on the KITTI stereo dataset, incorporating statistical significance tests to validate the superiority of our approach. We achieved an AP of 92.3% and an AR of 87.5%, significantly outperforming the Harris corner detector and SIFT.

This performance is further underlined by a detailed error analysis, revealing key strengths in complex scenarios such as occlusions and variable lighting. Additionally, our computational efficiency, averaging 0.034 s per frame, demonstrates practical applicability.

Extensive testing on the Middlebury and Karlsruhe datasets further substantiates our method's robustness, achieving APs of 87.2% and 84.6%, respectively. This consistent performance across diverse datasets highlights the adaptability of our approach (see Fig. 3).

Overall, our proposed approach achieved superior performance in terms of accuracy and computational efficiency compared to existing methods. These results demonstrate the effectiveness and practicality of our approach for vehicle detection in stereoscopic images.

In the context of object detection, AR (Average Recall) and AP (Average Precision) are commonly used evaluation metrics.

Average Recall (AR) is the average percentage of true positive detections across all recall values. Recall is the fraction of ground-truth objects that are correctly detected by the algorithm. AR provides an overall measure of how well the system is able to detect objects, regardless of their location, size, and orientation.

Average Precision (AP) is the area under the precision- recall curve, which measures how well the system is able to correctly detect objects while minimizing false positives. Precision is the fraction of true positive detections among all detections, while recall is the same as described above.

AP and AR are both useful evaluation metrics in object detection. AP places more emphasis on the precision-recall trade-off, while AR provides a measure of detection performance across all recall values.

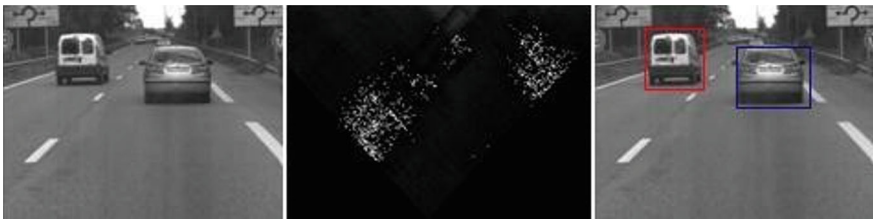


Fig. 3 Example results from our approach on the KITTI dataset. The red and the blue boxes indicate the detected vehicles

5 Discussion

In our analysis, we delve deeper into the implications of our AI-based approach for vehicle detection. The notable performance leap in our method, as evidenced by the superior AP and AR scores, is rooted in the innovative application of symmetry maps and advanced algorithms.

Our comparative discussion against state-of-the-art methods reveals key technical factors contributing to this enhancement.

While acknowledging limitations, such as reduced efficacy in crowded scenes, we propose future research directions, including algorithmic adjustments to address these challenges.

Conclusively, our approach not only demonstrates a significant advancement in vehicle detection but also lays the groundwork for future enhancements in driver assistance systems, potentially leading to safer and more reliable autonomous driving technologies.

6 Conclusion

This paper presents a novel vehicle detection approach in stereoscopic images, advancing the field with both technical innovation and practical application.

Our method significantly enhances driver assistance systems and shows promise for autonomous vehicle [7] development, as evidenced by robust performance in real-world scenarios like the KITTI dataset.

While acknowledging limitations, particularly in adverse weather and lighting conditions, this study sets the stage for future research to further improve and diversify this technology's applicability.

In essence, our work not only marks progress in vehicle detection but also lays the groundwork for future advancements in the field.

References

1. Alzubi, J.A., et al.: Distracted driver detection using compressed energy efficient convolutional neural network. J. Intell. Fuzzy Syst. **41**(3), 1–10 (2021)
2. Alzubi, O.A., Alzubi, J.A., Alweshah, M., Qiqieh, I., Al-Shami, S., Ramachandran, M.: An optimal pruning algorithm of classifier ensembles: dynamic programming approach. Neural Comput. Appl. **32**, 16091–16107 (2020)
3. Bradski, G.: The opencv library. Dr. Dobb's J.: Softw. Tools Prof. Program. **25**(11), 120–123 (2000)
4. Canny, John: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. **6**, 679–698 (1986)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), vol. 1, pp. 886–893. Ieee (2005)

6. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: The kitti vision benchmark suite **2**(5) (2015). http://www.cvlibs.net/datasets/kitti
7. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for autonomous driving? the kitti vision benchmark suite. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361. IEEE (2012)
8. Harris, C., Stephens, M., et al.: A combined corner and edge detector. In: Alvey Vision Conference, vol. 15, pp. 10–5244. Citeseer (1988)
9. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vis. **60**, 91–110 (2004)
10. Movassagh, A.A., et al.: Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. J. Ambient. Intell. Hum. Comput. (2020)
11. Shi, J., et al.: Good features to track. In: 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 593–600. IEEE (1994)

Enhancing Arabic Sentiment Analysis Using AraBERT and Deep Learning Models

Abderrahim Ouza, Ali Ouacha, Abdelhamid Rachidi, Mohamed El Ghmary, and Ali Choukri

Abstract The Arabic language possesses intricate morphology and relatively limited resources, and its syntax remains less explored in comparison to English. The complexity of this linguistic landscape presents distinct challenges for analyzing sentiments in Arabic texts. The array of opinions and emotions conveyed in Arabic is vast, demanding a more sophisticated methodology to accurately comprehend these nuances. This research is dedicated to sentiment analysis in Arabic texts through an innovative deep learning approach. In contrast to conventional techniques that might neglect the intricacies of Arabic, this strategy aims to harness the language's rich morphology and syntax to achieve more precise sentiment and emotion analysis. The main goal of this investigation was to devise a dependable and precise sentiment analysis system for Arabic texts, employing techniques from the realm of deep learning. The selection of AraBERT, a pre-trained model tailored for Arabic, and its amalgamation with diverse neural network architectures, sought to effectively exploit the distinctive linguistic traits of Arabic and enhance sentiment analysis. Within this study, a variety of models were developed, trained, and evaluated, each involving AraBERT and neural network designs like CNN, LSTM, and GRU. The models' performance was gauged based on precision, recall, F1 score, and accuracy. The results demonstrated that AraBERT with GRU and BI-GRU model fusion outperformed other techniques. These hybrid models performed well with accuracy of more than 93%, successfully capturing the subtle nature of Arabic emotions.

A. Ouza (✉)
Department of Computer Science, Faculty of Science, Ibn Tofail University, Kenitra, Morocco
e-mail: abderrahim.ouza@uit.ac.ma

A. Ouacha · A. Rachidi
Department of Computer Science, Faculty of Science, Mohammed V University, Rabat, Morocco
e-mail: a.ouacha@um5r.ac.ma

M. El Ghmary
Department of Computer Science, Faculty of Science, Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: mohamed.elghmary@usmba.ac.ma

A. Choukri
Department of Computer Science, Faculty of Science, Ibn Tofail University, Kenitra, Morocco
e-mail: ali.choukri@uit.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_15

1 Introduction

The growing significance of sentiment analysis in the realm of social media, with a particular focus on Arabic texts, stands out as an important observation. Social media platforms have given rise to an abundance of data reflecting opinions, viewpoints, and sentiments on various topics. The process of sentiment analysis plays a crucial role in extracting and categorizing these sentiments as either positive or negative, yielding valuable insights for businesses and organizations. Nevertheless, it is worth noting that the majority of research in this domain has predominantly concentrated on the English language, creating a research gap concerning Arabic sentiment analysis. With the surge of Arabic social media platforms and the increasing complexity of the language, the importance of sentiment analysis in the Arabic context as a research field has grown. Arabic presents specific challenges, such as intricate verb forms, dialectal variations, and distinct syntactic structures, necessitating tailored approaches and models for precise sentiment analysis. To tackle this challenge, this study utilizes the AraBERT embedding and explores various neural network architectures, including LSTM, BI-LSTM, GRU, BI-GRU, and CNN. More specifically, our ultimate goal in this study is to overcome the unique linguistic challenges of the Arabic context and achieve accurate emotional analysis. This research aims to provide methods and models that can be used to extract and classify emotions from Arabic texts, thereby providing valuable insights to businesses, organizations and researchers. In the second section, we will explore the various contributions that have been implemented in the field of sentiment analysis. In the third section, we will address the subject of deep learning in general by discussing the different machine learning methods used in this work. The fourth section focuses on the methodological approach adopted to develop efficient models for sentiment analysis. We will first address the question of the dataset used. Next, we will explore the AraBERT architecture in detail and describe the working process. In the fifth section, we dive into the essence of our study by presenting the working environment and the various experiments carried out as well as the results obtained. Finally, the last section presents a general conclusion.

2 Related Work

Recently, there has been a significant research focus on applying artificial intelligence techniques to sentiment analysis. Scholars have employed a variety of AI methodologies, including natural language processing (NLP), machine learning, and deep learning, to enhance the accuracy of sentiment analysis across different languages, including Arabic.

Sentiment analysis is a widely recognized challenge that involves determining the polarity of customers' opinions. The process of sentiment analysis includes extracting relevant features from a collection of texts, constructing a classification model, and evaluating its performance [26]. This typical procedural framework has been applied in various sentiment classification tasks, covering evaluations of pharmaceutical

products [28], online product reviews [25]. Moreover, in the domain of e-commerce, researchers have explored a sentiment analysis-based recommendation system integrated with the use of an ontology to better cater to customers needs [21]. The author [14] use transfer learning in conjunction with cutting-edge pre-trained language models like AraBERT and Flair embeddings to address the problem of aspect extraction (AE) in Arabic aspect-based sentiment analysis (ABSA). On the Arabic Hotel reviews dataset, their suggested model, BF-BiLSTM-CRF, achieves an amazing F1 score of 79.7%, showing a considerable improvement above baseline and alternative models. The [6] describes a system for predicting the sentiment of book reviews using preprocessing, counter vectorization, and machine learning methods. Reference [1] analyzes various methods for Arabic sentiment analysis, prioritizing models based on transformers that have strong F-scores. A natural language processing architecture that produces competitive performance on benchmark datasets is introduced in [2]. With a 90% accuracy rate, [4] produces the largest Arabic sentiment analysis dataset and investigates certain features for Arabic sentiment analysis. As a last example of continued innovation in the field, [11] introduces DE-CNN, a unique technique that performs better than existing algorithms in Arabic sentiment categorization.

Two notable approaches stand out:

Machine learning approach: The results of this study [24], which are supported by previous investigations into other techniques for sentiment analysis, consistently show that SVM is the most trustworthy classifier. The author of the study cited as [8] provided a technique emphasizing the value of sentiment analysis in Arabic social media material. They used machine learning classifiers like Logistic Regression, K-Nearest Neighbors, and Decision Tree and discovered that while K-Nearest Neighbors and Decision Tree perform well with smaller datasets like AJGT and ASTD, Logistic Regression gets greater accuracy with larger datasets. In the context of sentiment analysis in the Arab world, the author [3] has stated that this study highlights the importance of Twitter as a useful source for capturing public sentiment, especially on social concerns like unemployment in Saudi Arabia. In order to categorize feelings, they use machine learning, more especially the SVM and NB algorithms, highlighting the effect of n-gram representation on accuracy and detailing strategies for improving categorization through the use of a domain-specific knowledge base. According to the author's citation, [27] Arabic sentiment analysis has gotten relatively little study compared to English sentiment analysis at both the sentence and document levels. This research focuses on Arabic sentence-level sentiment analysis using 1000 tweets from Twitter and machine learning (ML) methods including Naive Bayes and SVM classifiers. Repetitive tweets, opinion spamming, and the prevalence of ambiguous feelings in test tweets present problems, nevertheless. By presenting a context-aware, deep-learning-driven approach for Persian sentiment analysis that explicitly targets movie reviews, the author [10] tackles the lack of attention paid to sentiment analysis in languages other than English. In a study comparing deep learning algorithms, LSTM is shown to perform better than other techniques including SVM, CNN, and logistic regression.

Deep learning approach: Notably, Heikal, Torki, and El-Makky in their paper [19] proposed a method combining CNN and LSTM models, attaining an F1 score of 64.46% on the ASTD dataset. Mohammed and Kora in their study [23] evaluated CNN, LSTM, and RCNN models, with LSTM achieving an average accuracy of 81.31%. Barhoumi et al. [7] evaluated various Arabic-specific embeddings to enhance sentiment analysis performance. Abu Kwaik et al. [22] surpassed benchmark models using the LSTM-CNN model with high accuracy rates. Elfaik and Nfaoui introduced a bidirectional LSTM model in their work [17] that outperformed existing approaches, utilizing deep contextualized embeddings. Furthermore, Fouad et al. [15] proposed ArWordVec word embedding models, demonstrating high efficiency in word similarity tasks and sentiment analysis. However, these studies failed to address certain weaknesses in their approaches. In this research, we aim to address this gap by utilizing the pre-trained AraBERT model, in combination with different neural network architectures such as LSTM, BI-LSTM, GRU, BI-GRU, and CNN. This novel approach possesses its distinctiveness and originality, as it harnesses the power of pre-trained language models specifically designed for Arabic text, along with diverse neural network architectures, to improve sentiment analysis accuracy for Arabic data. By utilizing deep learning techniques and word embeddings, the author [12] of this paper addresses the paucity of research on sentiment analysis in the Arabic language and produces outstanding results on benchmark Arabic datasets. On the HARD dataset with fastText embeddings, the proposed CNN model beats existing methods, obtaining a stunning 94.69% accuracy.

By integrating these advanced techniques, we seek to overcome the challenges presented by the complex structure of the Arabic language, dialect variations, and limited resources. Ultimately, our contribution will advance the field of sentiment analysis for Arabic text.

3 Deep Learning

Various deep learning [18] techniques are utilized for sentiment analysis in Arabic, such as employing the AraBERT word embedding method combined with techniques like CNN, LSTM, BI-LSTM, BI-GRU and GRU [16]. LSTM (Long Short-Term Memory) [20] is a type of recurrent network that can maintain long-term information through memory cells, enabling the capture of long-term dependencies in sequence data (refer to Fig. 1a).

GRU (Gated Recurrent Unit) is a simplified variant of LSTM, which uses gate mechanisms to control the information that is retained or forgotten, making the learning process more efficient [9] (refer to Fig. 1b).

BI-LSTM (Bidirectional LSTM) is an architecture that combines two LSTM models, one processing the sequence in the normal order and the other in the reverse order, allowing for consideration of both past and future context in sequence analysis.

BI-GRU (Bidirectional GRU) is similar to BI-LSTM, but uses GRU units instead of LSTM, enabling a better contextual understanding of textual data in both directions.

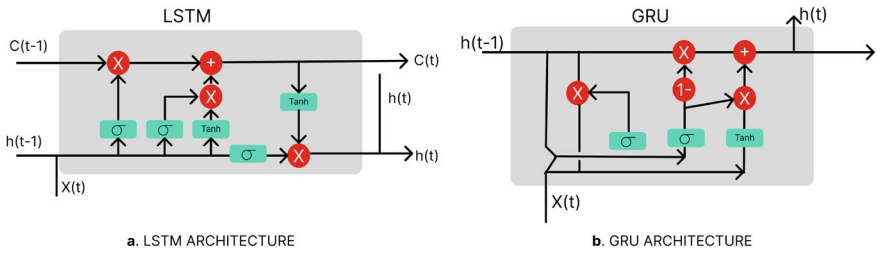


Fig. 1 LSTM and GRU architecture

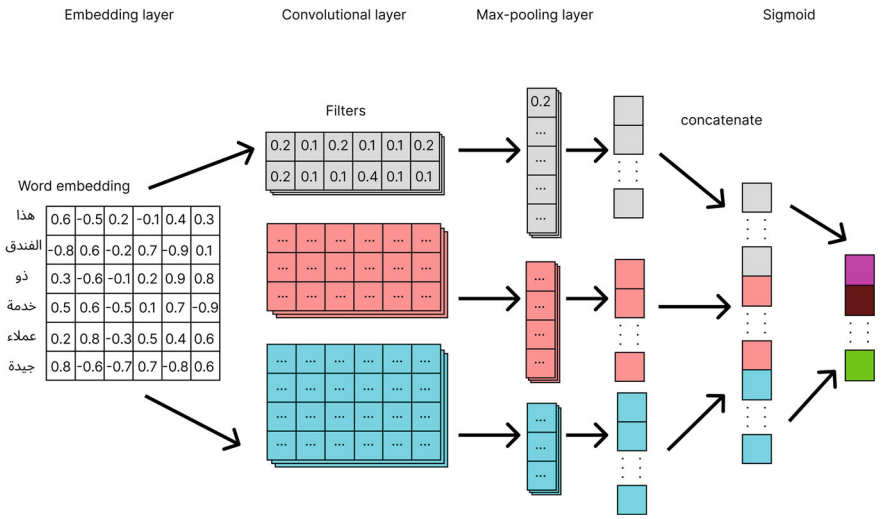


Fig. 2 CNN architecture

CNN (Convolutional Neural Network) is a model used in Natural Language Processing (NLP) to extract relevant features from textual data using convolution operations on word or character matrices [29] (refer to Fig. 2).

4 Proposed Methodology

4.1 Datasets

The dataset [13] was compiled in June and July 2016 from Booking.com and contains 93,700 hotel reviews in Modern Standard and Dialect Arabic. With 46,850 evaluations in both positive and negative categories, it is notable for its evenly distributed ratings, assuring linguistic variety and fair feeling class representation. Due to the inclusion of both Modern Standard and dialectal Arabic, this dataset stands out for

portraying the linguistic diversity found in actual Arabic literature. It also concentrates on hotel reviews, a field with specialized vocabulary and distinctive sentiment expressions.

4.2 Preprocessing

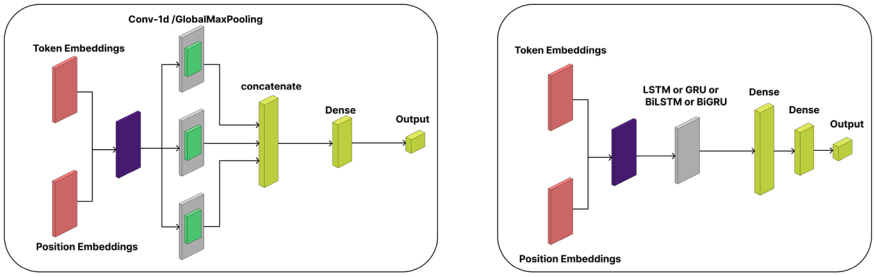
Reviews were pre-processed to remove words that had no bearing on sentiment analysis. This simplified word encoding and made it simpler to train the model. Diacritical marks, punctuation (including Arabic commas), multiple spaces, emojis, Arabic stop words, repeated characters, HTML tags, HTML entities, letters that aren't Arabic, and lines that contain the latter are all removed during the pre-processing stage. Arabic normalization and sequence vectorization with AraBERT were both done as well [5]. To ensure balanced representation, the best generalization, and to avoid model overfitting, the data was divided into three sets: 60% training, 20% validation, and 20% testing.

4.3 AraBERT

Using its pre-trained embeddings, the AraBERT model has been integrated into our deep learning architectures, effectively expressing Arabic words. The Arabic-specific AraBERT program was chosen since it can handle dialectal and linguistic complexity. Accurate contextual sentiment analysis in Arabic text has benefited from its shown performance in the language's text processing and contextualized embeddings.

4.4 AraBERT CNN Architecture

CNNs play a crucial role in extracting local features within text sequences, a valuable aspect for identifying emotions in comments. These convolutional neural networks use convolution filters to systematically scan text and extract relevant information. The AraBERT CNN model represents an architecture that seamlessly integrates the AraBERT model with 1D convolutional neural network (1D-CNN) layers, suitable for text processing. This model incorporates three distinct sets of convolutional layers operating in parallel, each using different kernel sizes and filters, followed by Global-max pooling layers. Subsequently, the outputs of these convolutional layers are concatenated. To improve the robustness of the model, fully connected layers are introduced, as well as a Dropout layer for regularization purposes. Finally, an output layer including a sigmoid activation function is added to enable binary classification (see Fig. 3a).



a. AraBERT CNN Architecture **b**. AraBERT Recurrent Neural Architecture

Fig. 3 AraBERT CNN and recurrent neural architecture

4.5 AraBERT Recurrent Neural Networks Architecture

RNNs offer a valuable capability for processing text in a sequential manner, allowing the model to grasp the inherent dependency relationships within Arabic text. This functionality contributes significantly to a more profound understanding of the subtleties and intricacies present in the expression of sentiments in the Arabic language. The architectural design of our model involves the integration of the AraBERT model with a recurrent neural network layer, such as LSTM, GRU, BiLSTM, or BiGRU. To introduce non-linearity and effectively capture intricate feature relationships, we augment the model with a dense layer and apply the ReLU activation function. Additionally, we incorporate a dropout layer to enhance model regularization and mitigate the risk of overfitting. Following this, another dense layer with ReLU activation is introduced, followed by an additional dropout layer. Finally, for the purpose of binary classification, an output layer comprising a single unit with a sigmoid activation function is utilized. This comprehensive architecture allows us to extract nuanced sentiment information from Arabic texts effectively (see Fig. 3b).

5 Experiment and Results

This work uses deep learning methodology to analyze sentiments in Arabic texts. The strategy used is based on the combination of various neural network designs with the pre-trained AraBERT model. The objective was to predict the emotions linked to each statement. Google Colab was the simulation environment used. According to the results, the AraBERT + GRU and AraBERT + BiGRU models performed the best, opening new avenues of investigation.

5.1 Simulation and Hyperparameters

The simulation environment utilized in this study was Google Colab Pro+, a platform offering enhanced features beyond the free version. These enhancements encompass increased computational capabilities, expanded memory capacity, extended execution times, and priority access to valuable resources. We meticulously adjusted several hyperparameters to optimize the model's performance, and the key settings are summarized in Table 1 below.

5.2 Simulation Results

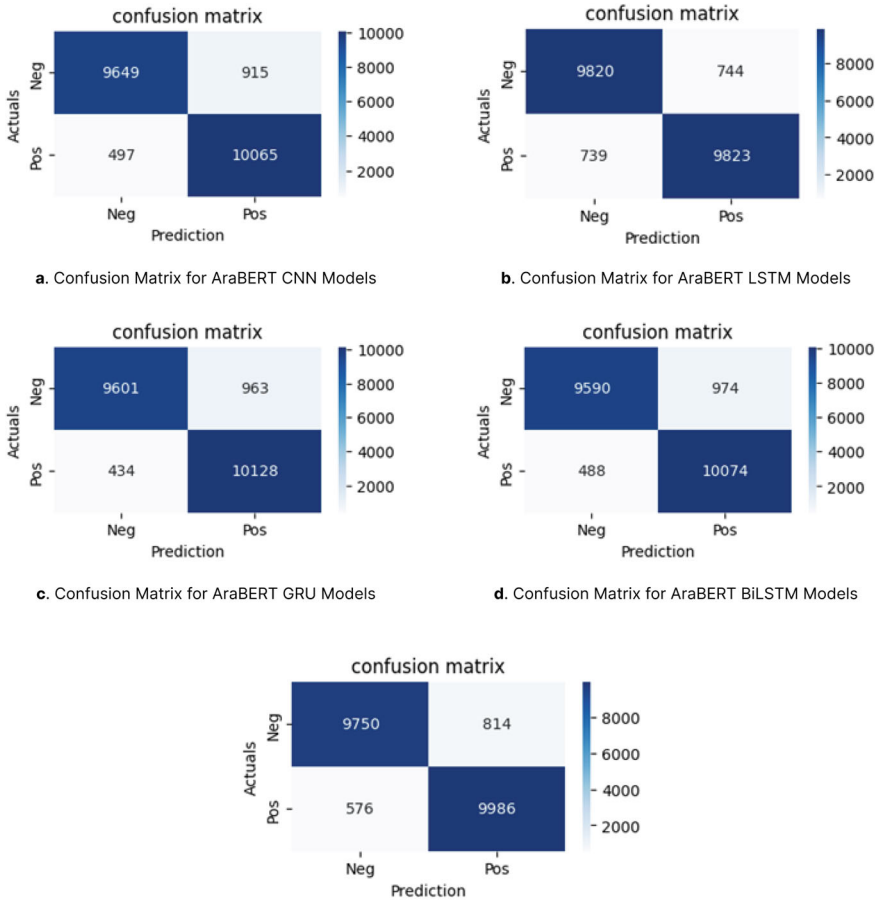
In Table 2 provides a detailed breakdown of the categorization outcomes from several models utilizing the Arabert language model, which is summarized in the confusion matrix graph (Fig. 4). The precision, recall, and F1 score for the "negative" and

Table 1 The hyperparameters used for models

Hyperparameter	Setting
Activation functions	RELU (hidden layers), Sigmoid (output layer)
Batch size	64
Dropout Rate	0.5
Encoder layers (AraBERT)	12
Learning rate	0.0001
Maximum sequence length	64
Number of epochs	20
Optimizer	Adam

Table 2 Summary of classification results

Model	Class	Precision (%)	Recall (%)	F1-Score (%)	Training time
AraBERT CNN	Negative	95.10	91.34	93.18	2 h 3 m 40 s
	Positive	91.67	95.29	93.45	
AraBERT LSTM	Negative	92.98	92.98	92.98	1 h 50 m 42 s
	Positive	92.98	92.98	92.98	
AraBERT GRU	Negative	95.67	90.88	93.22	2 h 22 m 4 s
	Positive	91.32	95.89	93.55	
AraBERT BiLSTM	Negative	95.16	90.78	92.92	1 h 52 m 14 s
	Positive	91.18	95.38	93.24	
AraBERT BiGRU	Negative	94.42	92.30	93.35	2 h 2 m 15 s
	Positive	92.46	94.55	93.49	



a. Confusion Matrix for AraBERT CNN Models

b. Confusion Matrix for AraBERT LSTM Models

c. Confusion Matrix for AraBERT GRU Models

d. Confusion Matrix for AraBERT BiLSTM Models

e. Confusion Matrix for AraBERT BiGRU Models

Fig. 4 AraBERT CNN and recurrent neural architecture

"positive" classes are summarized in the table along with the associated training times for each model.

The overview of the scoring models (Fig. 5) employed in this binary sentiment classification inquiry for Arabic texts includes the AraBERTCNN, AraBERT LSTM, AraBERT GRU, AraBERT BiLSTM, and AraBERT BiGRU models. The effectiveness of these models was assessed using a number of global metrics, including accuracy (ACCURACY), precision (PRECISION), recall (RECALL), and F1 score (F1-SCORE). The amount of time needed to train the models was also considered.

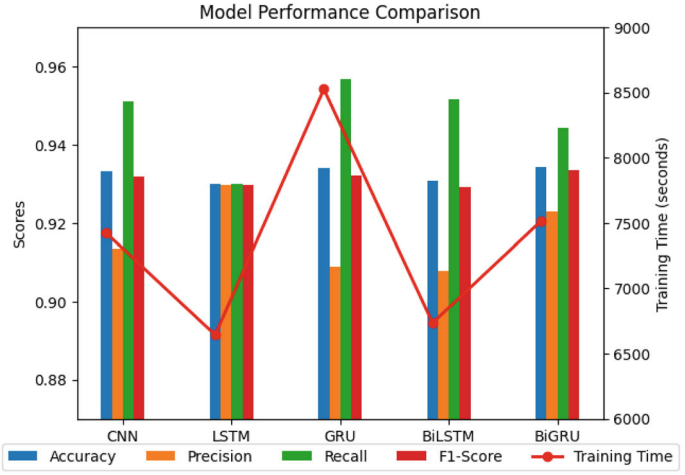


Fig. 5 AraBERT CNN and recurrent neural architecture

6 Conclusion

In this study, deep learning methods are used to examine sentiment analysis in Arabic texts. The major goal was to assess how well the AraBERT model performed when used in conjunction with different neural network architectures to predict the emotion of each comment. The findings reveal that the tested models-AraBERTCNN, AraBERT LSTM, AraBERT GRU, AraBERT BiLSTM, and AraBERT BiGRU-all performed well in classifying sentiment. AraBERT GRU and AraBERT BiGRU stood out among these models by earning the greatest marks for accuracy, recall, and F1, indicating that they are especially well-suited for the sentiment classification job for Arabic texts. The AraBERT + LSTM model stands out for its shorter learning time, though. These findings open the door to a range of fascinating possibilities, including the use of alternative evaluation metrics, comparison with existing Arabic sentiment analysis techniques, investigation of larger datasets or more challenging classification tasks, and extension of the models to other languages or comparable tasks.

References

1. Abu Farha, I., Magdy, W.: A comparative study of effective approaches for Arabic sentiment analysis. Inf. Process. Manag. (2021). https://doi.org/10.1016/j.ipm.2020.102438
2. Abu Farha, I., Magdy, W.: Mazajak: an online Arabic sentiment analyser. In: Proceedings of the Fourth Arabic Natural Language Processing Workshop, Florence, Italy, Association for Computational Linguistics, pp. 192–198 (2019). https://doi.org/10.18653/v1/W19-4621
3. Alwakid, G., Osman, T., Hughes-Roberts, T.: Challenges in sentiment analysis for Arabic social networks. Procedia Comput. Sci. **117**, 89–100 (2017). https://doi.org/10.1016/j.procs.2017.10.097

4. Aly, M., Atiya, A.: LABR: A Large Scale Arabic Book Reviews Dataset (2013). https://doi.org/10.13140/2.1.3960.5761
5. Antoun, W., Baly, F., Hajj, H.: AraBERT: transformer-based model for Arabic language understanding. In: Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection, Marseille, France, European Language Resource Association, pp. 9–15 (2020)
6. Azhaguramyaa, V.R., Janet, J., Madhavan, G.R., Balakrishnan, S., Arunkumar, K.: Sentiment analysis on book reviews using machine learning techniques. In: 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, pp. 1530–1534 (2022). https://doi.org/10.1109/ICACCS54159.2022.9785311
7. Barhoumi, A., Camelin, N., Chafik, A., Estève, Y., Belguith, L.: An Empirical Evaluation of Arabic-Specific Embeddings for Sentiment Analysis, pp. 34–48 (2019) https://doi.org/10.1007/978-3-030-32959-4_3
8. Bolbol, N.K., Maghari, A.Y.: Sentiment analysis of Arabic tweets using supervised machine learning. In: Proceedings of the 2020 International Conference on Promising Electronic Technologies (ICPET), Jerusalem, Palestine, pp. 89–93 (2020). https://doi.org/10.1109/ICPET51420.2020.00025
9. Chung, J., Gulcehre, C., Cho, K., Bengio, Y.: Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling (2014). arXiv:1412.3555
10. Dashtipour, K., Gogate, M., Adeel, A., Larijani, H., Hussain, A.: Sentiment analysis of Persian movie reviews using deep learning. Entropy **23**(5), 596 (2021). https://doi.org/10.3390/e23050596
11. Dahou, A., Elaziz, M.A., Zhou, J., Xiong, S.: Arabic sentiment classification using convolutional neural network and differential evolution algorithm. Comput. Intell. Neurosci. **2019**, 16 (2019). Article ID 2537689
12. Elhassan, N., Varone, G., Ahmed, R., Gogate, M., Dashtipour, K., Almoamari, H., El-Affendi, M., Al-Tamimi, B., Albalwy, F., Hussain, A.: Arabic sentiment analysis based on word embeddings and deep learning. Computers **12**(6) (2023). https://doi.org/10.3390/computers12060126
13. Elnagar, A., Khalifa, Y.S., Einea, A.: Hotel Arabic-reviews dataset construction for sentiment analysis applications. In: Shaalan, K., Hassanien, A., Tolba, F. (eds.) Intelligent Natural Language Processing: Trends and Applications. Studies in Computational Intelligence, vol. 740, pp. 35–52 (2018). Springer International Publishing. https://doi.org/10.1007/978-3-319-67056-0_3
14. Fadel, A., Saleh, M., Abulnaja, O.: Arabic aspect extraction based on stacked contextualized embedding with deep learning. IEEE Access **10**, 30526–30535 (2022). https://doi.org/10.1109/ACCESS.2022.3159252
15. Fouad, M.M., Mahany, A., Aljohani, N., Abbasi, R.A., Hassan, S.U.: ArWordVec: efficient word embedding models for Arabic tweets. Soft Comput. **24**(11), 8061–8068 (2020). https://doi.org/10.1007/s00500-019-04153-6
16. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016). http://www.deeplearningbook.org
17. Elfaik, Hanane, Nfaoui, El Habib: Deep contextualized embeddings for sentiment analysis of Arabic book's reviews. Procedia Comput. Sci. **215**, 973–982 (2022). https://doi.org/10.1016/j.procs.2022.12.100
18. Hameed, R., Abed, W., Sadiq, A.: Evaluation of hotel performance with sentiment analysis by deep learning techniques. Int. J. Interact. Mob. Technol. (iJIM) **17**, 70–87 (2023). https://doi.org/10.3991/ijim.v17i09.38755
19. Heikal, M., Torki, M., El-Makky, N.: Sentiment analysis of arabic tweets using deep learning. Procedia Comput. Sci. **142**, 114–122 (2018). ISSN 1877-0509. https://doi.org/10.1016/j.procs.2018.10.466
20. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735

21. Karthik, R.V., Ganapathy, S.: A fuzzy recommendation system for predicting the customers' interests using sentiment analysis and ontology in e-commerce. Appl. Soft Comput. **108**, 107396 (2021). https://doi.org/10.1016/j.asoc.2021.107396
22. Kwaik, K.A., Saad, M., Chatzikyriakidis, S., Dobnik, S.: LSTM-CNN deep learning model for sentiment analysis of dialectal Arabic. In: International Colloquium on Automata, Languages and Programming (2019). https://api.semanticscholar.org/CorpusID:203847790
23. Mohammed, A., Kora, R.: Deep learning approaches for Arabic sentiment analysis. Soc. Netw. Anal. Min. **9**(1), 52 (2019). https://doi.org/10.1007/s13278-019-0596-4
24. Nabil, M., Aly, M., Atiya, A.: ASTD: Arabic sentiment tweets dataset. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, Lisbon, Portugal, Association for Computational Linguistics, pp. 2515–2519 (2015). https://doi.org/10.18653/v1/D15-1299
25. Onan, A.: Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks. Concurr. Comput.: Pract. Exp. **33** (2020). https://doi.org/10.1002/cpe.5909
26. Sangeetha, J., Kumaran, U.: Sentiment analysis of Amazon user reviews using a hybrid approach. Meas.: Sens. **27**, 100790 (2023). ISSN 2665-9174. https://doi.org/10.1016/j.measen.2023.100790
27. Shoukry, A., Rafea, A.: Sentence-level Arabic sentiment analysis. In: Proceedings of the 2012 International Conference on Collaboration Technologies and Systems (CTS), Denver, CO, USA, pp. 546–550 (2012). https://doi.org/10.1109/CTS.2012.6261103
28. Suhartono, D., Purwandari, K., Jeremy, N.H., Philip, S., Arisaputra, P., Parmonangan, I.H.: Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews. Procedia Comput. Sci. **216**, 664–671 (2023). ISSN 1877-0509. https://doi.org/10.1016/j.procs.2022.12.182
29. Zhang, Y., Wallace, B.: A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification (2016). arXiv:1510.03820

Morphosyntactic Meaning of Arabic Words

Khaireddine Bacha

Abstract The morphological analysis of Arabic is reduced to a recognition of the components of the agglutinated graphic form and the identification of the role of each of the components. This paper presents a recent study aiming to develop a Part Of Speech (POS) tagger for Arabic language, it is organized as follows. In the first section, an overview of recent and a briefly introduce to different configurations and tested approaches in POS tagging. In the following section, we describe a robust approach that can greatly influence tagging performance to justify the methodology and appropriate parameter used to build our POS tagger.

Keywords Part-of-Speech tagging · POS tagger · Arabic · Disambiguation · Language Model

1 Introduction

The Arabic language and its dialects have attracted increasing attention in the Natural Language Processing (NLP) community. Projects aimed at revitalization, standardization and linguistic normalization have been launched to promote the use of this language and to contribute to its survival [2]. Starting from a minimalist morphological analysis, based on the principle that every Arabic linguistic form is translated into scheme and root, research will develop from the first work on the lexicon and morphology to the development of automatic analyzers, indexing systems, correctors, etc. Parts of speech (also known as POS, word classes, morphological classes or lexical tags) of the Arabic language have been an active research topic in recent years [7, 10].

K. Bacha (✉)
Laboratory LaTICE, University of Tunis, Tunis, Tunisia
e-mail: khairi.bacha@gmail.com

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_16

Many computational methods and algorithms for assigning POS to words have been used, including rule-based tagging as well as other methods such as transformation-based, memory-based tagging and also a hybrid tagging method [16, 19].

2 Critical of Existing and the Problem of Interoperability

Natural languages are ambiguous in nature. Ambiguity appears in different levels of natural language processing. The field of morphological analysis has been enriched in recent decades by the contributions of several researchers. Khoja [12] combines both statistical and rule-based techniques and uses a tag set of 131 derived from the BNC tag set. Yahya and Mohamed [21] presented an Arabic POS tagger that uses an HMM model, the evaluation showed an accuracy of around 96%. The Maamouri and Cieri system [14] achieved 96% accuracy, this is the automatic annotation output produced by AraMorph, which is Tim Buckwalter's morphological analyzer [4, 12]. Tlili Guiassa [22] uses a hybrid which consists of combining based-rules and memory-based learning methods. This system reports a performance of 85% [5].

Fatma al Shamsi and Guessoum [2] The HMM POS labeler was tested and achieved a peak performance of 97%. Part Of Speech Tagging.

The ambiguity is a central problem in the analysis morpho-syntactic of the Arabic. The analyzers are frequently confronted with situations of ambiguity at all levels of the analysis is at the level lexical, syntactic or semantic [13].

3 Part of Speech Tagging

The word graph in Arabic means a complex composition of objects. The basis of the word graph is called Minimum word, to which the addition of additional constituents considered an extension. When it receives external elements, it then becomes a maximum word composed of: proclitique, Prefix, basis, suffix, enclitique [14, 15] (Fig. 1).

3.1 Language Model

The language model, that is to say the component of the recognition system which is responsible for introducing the constraints imposed by the syntax of the language represented in the learning corpus, is currently based in recognition systems with large vocabulary the most efficient, on a probabilistic approach, compatible in this with the other components of the recognition system [13]. This

« Atatadhakkaronana» أَتَتَذَكَّرُونَنَا	
This word expresses the phrase in French: "Is that you remember us? " The segmentation of this word gives the following constituents: « na» نَا | « ona» ونَ | « tadhakkar» تَذَكَّر | « ta» تَ | « A» أ	
Antefix :	أ interrogation conjunction
Prefix :	تَ verbal prefix of the time of the unfulfilled
Schematic body:	تَذَكَّر derived from the root: ذكّر according to the schema تفعّل
Suffix :	ون verbal suffix expressing the plural
Post Fixed :	نا pronoun suffix noun complement

Fig. 1 Decomposition of the word graph in Arabic; "Atatadhakkaronana أَتَتَذَكَّرُونَنَا" "is that you remember us ?"

probabilistic language model is most often based on an empirical paradigm: a good estimate of the probability of a linguistic event can be obtained by observing this event on a training corpus of sufficient size [3]. The necessities induced by the labeling phase mean that the language model only takes into account the local constraints of the syntax, through so-called n-gram models, where the probability of a sentence is estimated from the conditional probabilities of occurrence of a word or a class of words. This approach is particularly interesting for its efficiency and robustness, but is limited to the modeling of local linguistic structures [20].

Currently bi- and trigram word models are commonly used in continuous speech recognition systems and in the automatic analysis of Arabic texts, the order of the models used going up to 4 (5-g model). In this context, we have built 3-g and MMC2 models for the Arabic language. For the construction of these models, we used:

The estimation probability:

$$P(w_i|et_i) = \frac{numberofoccw_i labeledbyet_i}{numberofoccdeet_i}$$

The transition probability of MMC2:

$$P(et_i|et_{i-1}) = \frac{numberofoccofsucc(et_{i-1}, et_i)}{numberofoccdeet_i}$$

And the Trigram transition probability a priori and posteriori:

$$\begin{cases} ifi > pso : P(et_i|et_{i-1}) = \frac{numberofoccofsucc(et_{i-1},et_i)}{numberofoccet_i} \\ ifi < pso : P(et_i|et_{i+1}) = \frac{numberofprevocc(et_i,et_{i+1})}{numberofoccet_i} \end{cases}$$

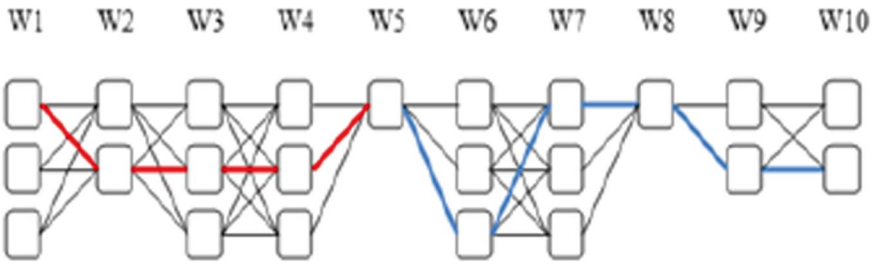


Fig. 2 Sentence decomposition

3.2 Disambiguation Approach

Labeling consists of assigning, for each of the identified lexical units, all possible morphosyntactic labels and fixing the most probable label. This operation, which is produced by a program called a labeler, can be done by consulting a lexicon where each form is followed by a list of categories either by morphological analysis or by combination.

According to the morphological analysis processing based on TELAMA finite state automata, we obtain as a result the set of possible labels for the graphic unit, and the list of labels. The morphological analyzer can also cause the assignment of a single label to the word, this is what we call the "unambiguous" word, characterized by a single label, then as a first labeling result we obtain the Fig. 2 next.

In the case of the existence of a single unambiguous word in the sentence to be processed, we fix the position of this word in a variable 'p' to use it later in our labeling system, and for the case of existence of different unambiguous words in the sentence fixed by the analyzer, we seek to select some, according to the emission probability $P(w_i|et_i)$ which has the equation:

$P(w_i|et_i) = \frac{numberofoccw_i labeledbyet_i}{numberofoccet_i}$This selected unambiguous word represents a reliable benchmark for labeling processing using the principles of Hidden Markov Model of degree 2 and N-grams.

3.3 N-grams

N-gram algorithms are widely used in signal processing and also in natural language processing, their use is based on the "simplification" hypothesis, given a sequence of K elements (K>n), the probability of the appearance of an element in position 'i' only depends on the previous 'n-1' elements. So we have:

$$P(et_i|et_1, \ldots, et_{i-1}) = P(et_i|et_{i-(n-i)}, et_{i-(n-2)}, \ldots, et_{i-1})$$

The studies carried out by Mars, [21], concerning the use of N-grams, represent a good statistical tool in the context of the automatic labeling of natural languages,

in the form of experiments carried out in his doctoral thesis, showing that the use of a trigram or an N-gram which has as N>3, generates excellent performance provided that it has the existence of a large learning corpus, if only to increase the number of transition sequences hence the increase in representative probability, moreover, Yamina [22] and BachaK et al. [23], reported that the higher order models converge and the calculation rate increases and reacts on the application d In a positive way, for this, we decided to apply the trigrams in TELAMA, which is represented by the equation:

$$P(et_i|et_1, \ldots, et_i) = P(et_i|et_{i-2}, et_{i-1}) with N = 3$$

For our TELAMA application, the starting position of the processing is the position of the unambiguous word which can be at the beginning, in the middle or at the end of the sentence, for this, we decompose this equation in two to adapt to our approach, hence what results in:

$$\begin{cases} P(et_i|et_1, \ldots, et_{i-1}) = P(et_i|et_{i-2}, et_{i-1}) i > p \\ P(et_i|et_i, \ldots, et_{i+1}) = P(et_i|et_{i+2}, et_{i+1}) i < p \end{cases}$$

The case i=p, does not require the realization of the principle of trigrams, to obtain the label of w_p, because this task is fixed from the outset during the sublabeling phase (Fig. 3).

Finally, we apply the probability of the sequence, which has the equation:

$$P(w_1, k) = P(w_1) \times P(w_2|w_1) \prod_{i=3}^{n} P(w_i|w_{i-2}, w_{i-1})$$

3.4 Hidden Markov Model

Given a sentence Ph={w_1,w_2, …,w_n}, this Markov Model tries to find the marker sequence; $T = \{et_1, et_2, \ldots, et_n\}$, Given a sentence Ph={w_1,w_2, …,w_n}, which maximizes the conditional probability P(T | Ph). We notice:

$$MaxT = \arg Max_T P(T|Ph)$$

With certain assumptions (independent hypotheses and Markov k=1 hypotheses, using binary successions), we have:

$$MaxT = \arg Max_T \prod_{i=1}^{n} P(w_i|et_i) \times P(et_i|et_{i-1})$$

With, $P(w_i|et_i)$ the emission probability is calculated in this way:

$$P(w_i|et_i) = \frac{numbereofoccdew_i labledbyet_i}{numberofoccW_i}$$

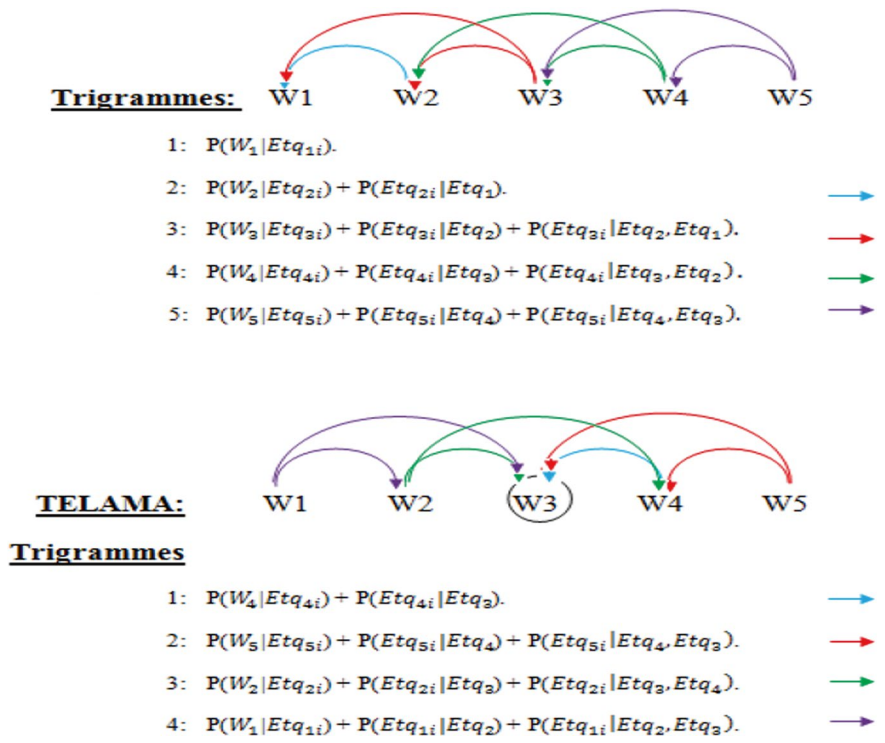


Fig. 3 Modification on Trigram

And $P(et_i|et_{i-1})$ the transition probability determined by:

$$P(et_i|et_{i-1}) = \frac{numberofoccofsucc(et_{i-1}, et_i)}{numberofoccet_i}$$

For TELAMA, the application of MMC requires the execution of the transition equation in two directions, depending on the precedence and the succession due to the use of the position of the unambiguous word 'p' during operation:

$$\begin{cases} sii > palors : P(et_i|et_{i-1}) = \frac{numberofoccofsucc(et_{i-1},et_i)}{numberofoccet_i} \\ sii < palors : P(et_i|et_{i+1}) = \frac{numberofprevocc(et_i,et_{i+1})}{numberofoccet_i} \end{cases}$$

The probabilities of emissions and transmissions in the two approaches Trigrams and MMC2, are calculated at the beginning by the language model which calculates the probabilities and the guards, when an approach requires a probability, all you have to do is make a call to the model to obtain a result, this is to save time during execution (Fig. 4). Statistical approaches to disambiguation need data from learning to model the language. The analysis of a text is based on techniques statistics and learning data. By observing the Arabic language and analyzing the

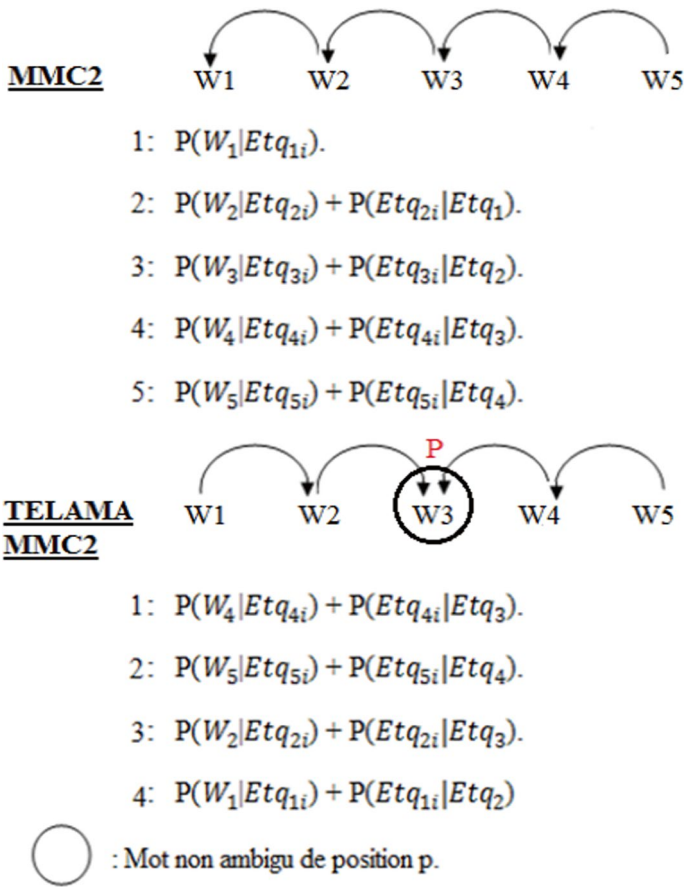


Fig. 4 Modification on MMC degree 2

results of the morphological analyzer, we notice that the statistical approach has problems distinguishing certain pairs of labels. To really see the effect of each label on the performance of a parser, the total occurrences of each label in the test corpus are calculated and verified.

4 Architecture of POS Tagger

This figure (Fig. 5) gives the architecture of our POS tagger for Arabic language. morphological system of analysis of Arab texts. The system comprises three modules: the morphological analyzer making it possible to assign with each lexeme one or several labels, the theory of probability which make it possible to assign

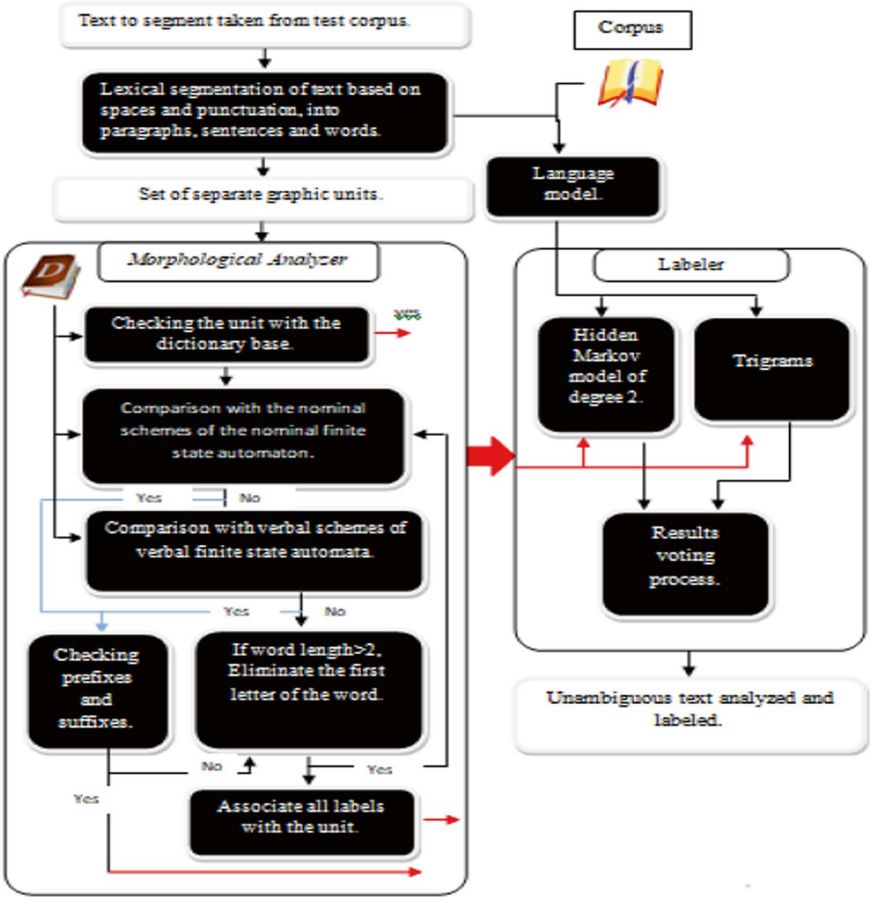


Fig. 5 Our Arabic POS tagger architecture

a probability with each potential sequence of labels, and modulate it clarification making it possible to calculate the most probable sequence of labels for each sequence of the text to be analyzed. This system functions according to well-defined relationships between the parts that compose it: segmentation, morphological analysis and labeling. These relationships are created to help the system using the training corpora and dictionary provided as resources, and obtain efficient and robust analysis with a low error rate and reliable and accurate results. In support of this and other works, we have described an architecture for our POS tagging system for Arabic language [16]. This architecture defines two components for disambiguation systems.

POS Tagger, it is implemented as an analysis module. (Fig. 5) illustrates the overall architecture, it attempts to choose the correct tag or lexical category, from a list of tags, for each words in a given text. This operation is achieved by the use

Table 1 Part of the confusion matrix

Tag	VB (%)	NN (%)	DT (%)
VB	97.30	01.80	00.90
NN	03.47	92.20	04.33
DT	02.47	4.36	93.17

of Viterbi Algorithm and all needed files like Map file and Language Model file computed in training level.

The voting process is synchronized so that it can, at the end of storing the results after the execution of the MMC of degree 1 and Trigrams, vote among the two solutions, which corresponds to the word. We are then faced with two cases:

- If the two results obtained are identical, and the label assigned to the word appears among all the labels provided by the morphological analyzer, then we confirm the label for the word in question and we archive the result in a performance database.
- If the two results obtained and proposed by the analyzer are completely different, we are faced with two possibilities:

Taggers are often evaluated by comparing them with a hand tagged test set, based on accuracy. We have prepared and tagged a test corpus (TestSet). Experimental results have been evaluated through standardize formulation: Precision, Recall and F-measure [8].

- *Precision = Correct number of token tag pair occurrence/Total number of token tag pair.*
- *Recall = Correct number of token tag pair occurrence/Number of correct token tag pair that is possible.*

$$F = 2 \cdot \frac{\text{Precision} \cdot \text{recall}}{\text{Precision} + \text{recall}}$$

The proposed POS tagger has been tested and has achieved a state of the art performance of 96% which is very encouraging.

A confusion matrix contains information about correct and predicted tag done by the POS tagger. Performance of such systems can be represented using the following matrix. The following Table 1 shows a part of the confusion matrix.

5 Conclusion

The morphological analysis of Arabic focuses, as for other languages, on word formats. In Arabic, morphological analysis is all the more important as the words are very clumped together, that is to say that each graphic form can contain a concatenation of several segments which each form a word or a lexical unit.

The development of an automatic morphological parser (POS Tagger) requires either a complete set of linguistic rules or a large, broad coverage tagged corpus.

To sum up, Arabic language POS marking is not an area for the faint of heart or easily depressed. It is sometimes very difficult, even for a human, to identify the correct label for a given word.

We also conclude that choosing the right set of labels is particularly useful for developing a morphological analyzer, it can influence its performance. This choice is due to the use of our morphological analyzer in an Arabic language learning platform which requires the greatest information on the words of the text to allow the automatic generation of educational activities.

References

1. Hifny, Y.: Recent advances in Arabic syntactic diacritics restoration. In: ICASSP International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 7768–7772. IEEE (2021)
2. Bacha, K.: Toward a model of statistical machine translation Arabic-French. In: International Conference on Advanced Learning Technologies and Education. ICALTE, Hammamet, Tunisia (2014)
3. Bacha, K.: TELA: Toward Environmental Learning Arabic. In: The International Conference on Artificial Intelligence (CIIA 11), p. 6. WORLDCOMP 11, Las Vegas, Nevada, USA. http://cerc.wvu.edu/download/WORLDCOMP%2711/2011%20CD%20papers/EEE4685.pdf) (2011)
4. Statista Most common languages used on the internet 2020. Online source. Accessed 8 Aug 2021
5. https://www.statista.com/statistics/262946/share-of-the-most-common-languages-on-the-internet/ (2020)
6. Bacha, K.: Design and implementation of a model of segmentation environmental Computer Assisted Learning "TELA". CIIA'sOusse, Tunisia (2014)
7. Mona, D., Kadri, H., Daniel, J.: Automatic tagging of Arabic text: from raw text to base phrase chunks. In: HLT-NAACL, pp. 149–152 (2004)
8. Freeman, A.: Brill's POS tagger and a morphology parser for Arabic. In ACL'01 Workshop on Arabic Language Processing (2001)
9. Maulud, D.H., Ameen, S.Y., Omar, N., Kak, S.F., Rashid, Z.N., Yasin, H.M., Ahmed, D.M.: Review on natural language processing based on different techniques. Asian J. Res. Comput. Sci. 1–17 (2021)
10. Saidi, M.A.: Support vector machine approach for examining Arabic content reports and classifying the part of speech tagger. Available at SSRN 3573555 (2020)
11. Karin, C.R.: A Reference Grammar of Modern Standard Arabic, p. 736. Cambridge University Press. ISBN: 0521777712 (2005)
12. Bacha, K.: Designing a model of Arabic derivation, for use in computer assisted Teaching. In: International Conference on Knowledge Engineering and Ontology Development. KEOD, Barcelona, Spain (2012)
13. Yousif, J., Al-Risi, M.: Part of speech tagger for Arabic text based support vector machines: a review. ICTACT J. Soft Comput. **10** (2019)
14. Alkishri, W., Almutoory, K.:Cloud computing architecture for tagging Arabic text using hybrid model. Speech Signal Process. **1**, 181–184 (2021)
15. Bacha, K.: Help system for creating educational resources for Arabic. Int. J. Knowl. Syst. Sci. (IJKSS) **9**(3) (2018)

16. Bacha, K.: Machine translation system on the pair of Arabic/English. In: International Conference on Knowledge Engineering and Ontology Development. KEOD, Barcelona, Spain (2012)
17. Bacha, K.: Designing a model combination of Arabic, for use in computer assisted teaching. In: World Congress on Computer Applications and Information Systems, pp. 1–7, (2014)
18. Bacha, K.: M contribution to a new approach to analyzing Arabic words. ICCCI **2**, 46–57 (2019)
19. Ney, H., Essen, Kneser, R.: On structuring probabilistic dependencies in stochastic language modeling. Comput. Speech Lang. **8**(1), 1–28 (1994)
20. Stolcke, A.: SRILM—an extensible language modeling toolkit. In: Proceeding of International Conference in Spoken Language Processing, Denver, Colorado (2002)
21. Yahya, O., Mohamed, E.: Statistical part-of-speech tagger for traditional Arabic texts. J. Comput. Sci. **5**(11), 794–800 (2009)
22. Yamina, T.G.: Hybrid method for tagging Arabic text. J. Comput. Sci. **2**(3), 245–248 (2006). ISSN 1549-3636
23. Bacha, K., Jemni, M., Zigui, M.: Toward a learning system based on Arabic NLP tools. Int. J. Inform. Retrieval Res. (IJIRR) **6**(4) (2018)

Semantic Similarity Between Arabic Questions Using Support Vector Machines and Hungarian Method

Samira Boudaa, Anass El Haddadi and Tarik Boudaa

Abstract Semantic tasks in Natural Language Processing (NLP), such as Semantic Textual Similarity and Textual Entailment, play a crucial role in numerous advanced NLP applications. Among these tasks, Semantic Question Similarity is particularly important for improving the performance of systems in areas such as Question Answering, chatbots, automated customer support, and Community Question Answering platforms. In this work, we propose an approach to deal with Semantic Question Similarity using the Hungarian method and Support Vector Machines. Our methodology builds upon an existing system originally designed for Recognizing Textual Entailment in Arabic, incorporating improvements and specific adaptations to address the particular characteristics of the Question Similarity task. The evaluation of our approach was performed on an existing dataset and has yielded encouraging results.

Keywords Semantic Textual Similarity · Question Similarity · Arabic Natural Language Processing · Machine Learning

1 Introduction

Semantics is a subfield of linguistics that studies the meaning of words, sentences, texts, or conversations. Various challenges arise in representing meaning, including the impact of the context on interpretation, ambiguity, and compositionality. In Natural Language Processing (NLP), semantic tasks such as Semantic Textual

S. Boudaa (✉) · A.E. Haddadi · T. Boudaa
National School of Applied Sciences, Abdelmalek Essaadi University, Al Hoceima, Morocco
e-mail: samira.boudaa@etu.uae.ac.ma

A.E. Haddadi
e-mail: a.elhaddadi@uae.ac.ma

T. Boudaa
e-mail: t.boudaa@uae.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_17

Similarity (STS), Recognizing Textual Entailment (RTE) (also known as Natural Language Inference), and Paraphrase Detection (PD) have recently attracted a lot of attention in research. These tasks are incorporated in many high-level NLP applications, including Question Answering Systems (QA) [1], Summarization [2], Information Retrieval (IR) [3], Machine Translation (MT) [4], and Plagiarism Detection (PD) [5], among others.

Word embedding has emerged as one of the techniques contributing high performances in NLP semantic tasks. Training word embedding models is still a very active research area with various embedding algorithms and parameters. Predicting the most pertinent word embedding for a specific task remains a challenging endeavor. Another important application of the semantic tasks is in the evaluation of word embedding models. Several research works have evaluated different embedding models based on NLP semantic tasks (e.g. [6, 7]).

Semantic Textual Similarity (STS) measures the similarity in meaning between two given texts. We note that this task is different from Lexical Similarity. While Lexical Similarity evaluates the similarity between two texts based on their word overlap, Semantic Similarity measures the similarity based on meaning rather than lexicographical similarity.

Other semantic tasks related to STS include Recognizing Textual Entailment (RTE), Paraphrase Detection (PD), and Semantic Relatedness. RTE is a unidirectional relation between two text fragments, it is a task that aims to decide whether the meaning of one text called Hypothesis can be inferred (entailed) from another text called Text [8]. PD, on the other hand, is a bidirectional relation that aims to identify the presence or absence of semantic equivalence between two statements. Typically, both tasks are modeled as binary supervised classification problem in which classifiers are tasked with determining whether two texts are paraphrases in the case of PD and whether one text can or cannot be inferred from the other in the case of RTE. While semantic relatedness of two concepts is representing the degree of their closeness or association, it is important to note that this relation does not necessarily imply semantic similarity; two concepts can be related but not similar (e.g. “fast food” and “heart disease” are related but not semantically similar).

Due to its particularities, the Arabic language remains challenging for NLP applications. While several research works and tools exist for English STS, there is still a limited amount of work in this field for Arabic. In the context of this study, we present an approach to deal with a specific case of Semantic Textual Similarity named Semantic Question Similarity where the system is asked to judge whether two questions are semantically similar or not; hence, the output is binary (1 if the input questions are similar and 0 if not). Semantic Question Similarity is especially important in question-driven applications such as building Question Answering systems, chatbots, and the optimization of Community Question Answering platforms. Its primary role lies in identifying similar questions, thereby enabling the provision of relevant and accurate answers, contributing to enhancing user interactions and knowledge sharing.

The rest of this paper is organized as follows: Sect. 2 presents an overview of the main existing work related to STS for Arabic and the related work. Section 3

explains the proposed approach in this work. Section 4 describes experiments and discusses the obtained results. Finally, conclusion and directions for future work in Sect. 5.

2 Related Work

STS was first presented at the International Workshop on Semantic Evaluation (SemEval 2012). It aims to automatically evaluate the semantic similarity between a pair of texts on a given scale (e.g. from 0 to 5) [9]. Subsequently, many variations of this task are proposed (e.g. cross-lingual and multilingual STS [10, 11]). While several approaches and datasets have been proposed for the English language, there have also been some attempts for Arabic. For instance, Almarsoomi et al. [12] followed a knowledge-based approach by proposing an algorithm for calculating word similarity using Arabic WordNet hierarchy and Li similarity measure [13], which calculates the similarity between two words, w1 and w2, as a function of the length of the shortest path between w1 and w2 and the depth of their Lowest Common Subsumer (LCS) in a lexical hierarchy. Moreover, Wali et al. [14] suggested a hybrid similarity measure that combines lexical similarity, semantic similarity using WordNet, and syntactico-semantic similarity using VerbNet. Recently, Word Embedding has attracted great interest in many NLP applications as it improves capturing the syntactic and semantic properties of words. Several existing systems for STS have used Word Embedding techniques. In this direction, Nagoudi and Schwab [15] proposed a word embedding-based system to measure semantic similarity for sentences, the system results were improved when Inverse Document Frequency (IDF) weighting or Part-Of-Speech (POS) tagging is applied. Another word embedding based system was presented by Nagoudi et al. [16] for the measuring of the Semantic Similarity between Arabic-English sentences. They presented two methods, the first method consists of words alignment combined with weighting functions presented previously in [15]. In the second method, for every pair of aligned words w and w' in the examined sentences, they extract the Bag-Of-Word (BOWw and BOWw') and they incorporate the Jaccard similarity between the two BOWs in the evaluation of the semantic similarity between the pair of sentences. Ismail et al. [17] also proposed an alignment and vector space-based approach to measure the degree of similarity between two sentences. First, using the lemma form of each word to generate its word space from an existing vector model. Then, constructing a word alignment matrix by coupling each word from S1 with the most suitable word from S2 based on the length of the intersection of their word spaces.

Some efforts were concentrated on depicting Semantic Question Similarity in Arabic. For instance, Al-Theiabat and Al-Sadi [18] used different deep learning models and achieved the best results by fine-tuning the multilingual BERT model and applying sentence pair classification task on Arabic questions. Another method was proposed by Fadel et al. [19] using word embeddings obtained

through the adoption of Arabic ELMo model [20]. Subsequently, a merging function was applied to the vectors representing each question pair in order to obtain a single representation vector. This vector serves as input for the Deep Neural Network layer, predicting whether the question pair is similar or not. Another Deep learning method was presented by Othman et al. [21] for similar question retrieval. They proposed an Attentive Siamese Long Short-Term Memory (LSTM) approach using word embedding learning and Manhattan Long Short-Term Memory (MaLSTM). Moreover, a rule-based approach with a supervised learning algorithm was proposed by Daoud [22]. The method consists on normalizing the questions and measuring the similarity based on a set of features. Those features include the question word similarity based on question scope (Time Factoid, Location Factoid, etc.), the start similarity, which is the similarity between the first words in q1 and q2, and the end similarity, which is the similarity between the last words in q1 and q2.

The present work is an extension of the methods presented in [23] that proposed an alignment-based approach for Recognizing Textual Entailment. The main stages of this approach are:

Preprocessing: The first stage involves the representation of the input texts T and H as two sequences of components belonging to one of the following types: Named Entities (NE), Temporal Expressions (TempEx), Number/Countable pairs (NC), ordinary words (or sequence of ordinary words). For this purpose, several existing NLP tools for Arabic are used, namely: Farasa[1] and Aratimex [24].

Alignment: In the second stage, the system seeks to find the best alignment between these representations by modeling the problem as an assignment problem that can be solved using existing mathematical algorithms such as the Hungarian method [25] or Jonker-Volgenant-Castanon (JVC) algorithm [26, 27].

Classification: The final decision made by the system is based on an SVM classifier where the score of the assignment is considered the main feature for textual entailment classification.

3 The Proposed Approach for Semantic Question Similarity

Many semantic tasks present a certain level of closeness. Thus, the approaches used to handle them are also somewhat similar. In this regard, our objective is to extend and generalize the alignment-based RTE system presented in the previous section to be applicable to Semantic Question Similarity.

For the preprocessing stage, we carried out the same steps as in the previous system namely: normalization of temporal expressions, numbers, named entity

[1] https://farasa.qcri.org/.

extraction and enrichments from knowledge resources (wikidata), and extraction of the synonyms of words from Arabic WordNet [28]. For the basic NLP subtasks, we used Farasa and Stanford CoreNLP.[2] The result of this step is a representation of the text as a sequence of components belonging to one of these types: Word, Number/Countable, Named Entity, and Temporal Expression.

Formally, given two input texts T1 and T2 we note respectively their rich representations containing the result of the preprocessing stage as follows:

$$R(T1) = \{t1_i / i \in \{1, \ldots, m\}\}$$

$$R(T2) = \left\{t2_j / j \in \{1, \ldots, n\}\right\}$$

While:

$t1_i$: the ith component of T1 with associated enrichment information.

$t2_j$: the jth component of T2 with associated enrichment information.

In the second stage, the system aligns the representations of T1 and T2 using the same algorithm as in the previous work with some important additional improvements. This algorithm is based on modeling the alignment problem an assignment problem that we can solve using the Hungarian method. These improvements and extensions are presented in the rest of this section.

An important point that influences semantic relations is stop word handling. Indeed, some stop words are important for some tasks, for instance, question words (i.e. 'why', 'who', etc.) are important for semantic question similarity, and negation words are very important for textual entailment. While the previous system ignores stop words, in our case we do not filter question words that have an important role in question similarity. Table 1 lists the question words in Arabic.

We used one-hot encoding to encode the type of the question based on the question word found at the beginning of the question.

One important property of the semantic relations between two texts is symmetry. Symmetry means that for all texts X and all texts Y, X *relatesTo* Y implies Y *relatesTo* X, where *relatesTo* is a semantic relation. STS is a symmetric relation where the pair of texts are judged similar if the information or the meaning embedded in both texts is the same. We defined two penalty parameters that guide the behavior of the system in the case where one text contains more or less information than the other. These parameters were also defined in the context of the previous work, but they were fixed empirically and do not depend on the importance of the words. In this work, we tuned their values using a small-sized dataset and we redefined them in a way so that they depend on the importance of the

[2] https://stanfordnlp.github.io/CoreNLP/.

Table 1 Arabic question words

Arabic word	Transliteration	Translation
من	mn	Who; whom
ما	mA	What
بم	bm	With what
متى	mtY	When
أين	Oyn	Where
كم	km	How much; how many
كيف	kyf	How
أي	Oy	Which; what
هل	hl	Introduces yes/no question
أ	O	Introduces yes/no question
أيان	OyAn	When
أنى	OnY	How; when; whence
ماذا	mA*A	What
لماذا	lmA*A	Why
لم	lm	Why
مم	mm	Of what

words (their TF-IDF). This setting allows flexible penalty values depending on the targeted task as shown below:

If |R (T1)| > |R (T2)| we complete R(T2) with fictive components and we consider that the assignment cost between a fictive component $t2_i$ of R(T2) and any component of R(T1) is equal to a real parameter $\alpha * tfidf(t1_i)$. If |R (T2)| > |R (T1)| we complete R(T1) with fictive components. The assignment cost between a fictive component |R (T2)| If |R (T2)| > |R (T1)| we complete R(T1) with fictive components. The assignment cost between a fictive component |R (T1)| we complete R(T1) with fictive components. The assignment cost between a fictive component $t1_i$ of R(T1) and any component of R(T2) is equal to a real parameter $\beta * tfidf(t2_i)$.

Another important improvement is at the similarity function level. Instead of using just knowledge resources and ontologies such as WordNet to compute the similarity between words, we used a more efficient technique by combining the usage of these knowledge resources and word embedding using the following algorithm:

Inputs: C_1, C_2: Components; WE: Word Embedding model.
OOV = Words out of vocabulary for WE.
$Synonyms(X)$ = the set of synonyms of component X.
$Vect(WE, X)$ = the vector representation of component X in WE
IF C_1, C_2 are Temporal expressions Or Numbers and $C_1 = C_2$:
return 1
IF C_1, C_2 are words or named entities
and *areSynonyms* (C_1, C_2): *// Using WordNet or wikidata*

return 1
$\overrightarrow{C_1} \leftarrow Vect(WE, C_1)$
$\overrightarrow{C_2} \leftarrow Vect(WE, C_2)$
IF $\overrightarrow{C_1} = \overrightarrow{0}$ and $(\exists synoC_1 \in Synonyms(C_1) \cap \overline{OOV})$:
$\overrightarrow{C_1} \leftarrow Vect(WE, synoC_1)$
IF $\overrightarrow{C_2} = \overrightarrow{0}$ and $(\exists synoC_2 \in Synonyms(C_2) \cap \overline{OOV})$:
$\overrightarrow{C_2} \leftarrow Vect(WE, synoC_2)$
IF $\overrightarrow{C_1} \neq \overrightarrow{0}$ and $\overrightarrow{C_2} \neq \overrightarrow{0}$
return $\cos(\overrightarrow{C_1}, \overrightarrow{C_2})$
return 0

For the classification decision, we employed a supervised machine learning approach using a range of classifiers, including Random Forest, Extra Trees, AdaBoost, Logistic Regression, Gradient Boosting and SVM. Among these classifiers, SVM demonstrated the best performance, and thus, we present results exclusively from the implementation using SVM.

The pairs of texts are input into the SVM classifier as feature vectors containing the Alignment score and the set of features listed below:

- F0 = Alignment score Knowledge Base only
- F1 = Alignment score Knowledge Base only + WE
- F2 = Number of different words without taking into account their TF-IDF
- F3 = Number of different words taking into account their TF-IDF
- F4 = Longest Common Subsequence
- F5 = Number of common named entities
- F6 = Number of different named entities
- F7 = Lexical overlap
- F8 = The question type of the Text1
- F9 = The question type of the Text2

Figure 1 summarizes the architecture of the final system.

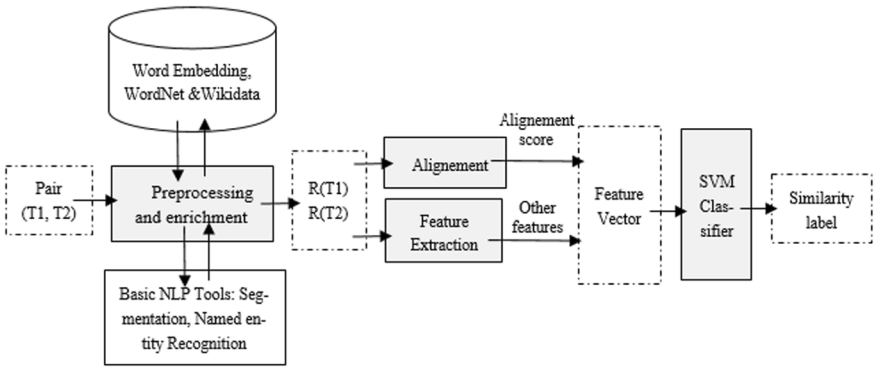


Fig. 1 The proposed system design

4 Evaluation and Results

To evaluate our approach we used the training nsurl-2019-task8 dataset [29]. It contains 11,997 pairs of questions, among them 5,397 (45%) are annotated with "0" (no semantic similarity), and 6,600 (55%) are annotated with "1" (semantically similar).

We split the dataset to train and test sets: 75% of the pairs are used for training and model hyperparameters tuning, while the 25% remaining data is used for testing. We conducted our experiments using scikit-learn package [30]. The SVM classifier uses RBF kernel, and the hyperparameters Gamma and C are tuned using tenfold cross-validation. For word embedding, we used the embeddings proposed by Altowayan and Tao [31].

The aligner parameters α and β are tuned using a small amount of the data. The values used for these parameters in the experiments are $\alpha = \beta = 0.8$.

We experimented with different sets of features in order to show the importance of the alignment score on one hand, and the combination of word embedding and knowledge bases to compute similarity at word level on the other hand. The results are reported in Table 2.

Our system has obtained encouraging results in its best performing run with an accuracy of 78.5%. We note the difference in term of accuracy between run 2 (70.967%), where only the question type and the alignment score without word embedding were used, and run 3 (73.8%) where we used word embedding. This shows that word embedding improves the results for depicting Semantic Question Similarity when it is combined with classical knowledge resources. The best performance is obtained in run 1 when alignment and Word Embedding are combined with other features that characterize the semantic similarity. Moreover, we notice that our system performs better in the case of no-labeled question pairs than the yes-labeled ones with the highest f1-score of 81.764%.

Table 2 Results and evaluation settings

Runs	Accuracy (%)	Similarity label	Precision (%)	Recall (%)	F1-score (%)
Run 1: using all features except F0	78.500	No	76.630	87.636	81.764
		Yes	81.671	67.333	73.812
Run 2: F0, F8, F9	70.967	No	73.254	74.364	73.805
		Yes	68.075	66.815	67.439
Run 3: F1, F8, F9	73.800	No	71.450	87.212	78.548
		Yes	78.600	57.407	66.353

5 Conclusion

In this paper, we described the Semantic Textual Similarity task and presented our work for measuring a specific case of Semantic Textual Similarity named Semantic Question Similarity. We used existing NLP tools and knowledge resources for Arabic and combined them with word embedding, alignment, and a set of features that characterize the semantic similarity of a pair of questions. Our system achieved encouraging results with an accuracy of 78.50%.

In future work, there is potential to adapt the proposed method for assessing semantic similarity between longer texts that are not necessarily questions. This adaptation can be evaluated on different datasets to enhance its generalizability.

References

1. Harabagiu, S., Hickl, A.:, Methods for using textual entailment in open-domain question answering. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the ACL—ACL '06, Sydney, Australia, pp. 905–912 (2006). https://doi.org/10.3115/1220175.1220289
2. Lauscher, A., Glavaš, G., Eckert, K.: University of mannheim@ clscisumm-17: citation-based summarization of scientific articles using semantic textual similarity. In: CEUR workshop proceedings, vol. 2002, pp. 33–42 (2017)
3. Boltužić, F., Šnajder, J.: Identifying prominent arguments in online debates using semantic textual similarity. In: Proceedings of the 2nd Workshop on Argumentation Mining, Denver, CO, pp. 110–115 (2015). https://doi.org/10.3115/v1/W15-0514
4. Marton, Y., Callison-Burch, C., Resnik, P.: Improved statistical machine translation using monolingually-derived paraphrases. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pp. 381–390 (2009)
5. Hunt, E. et al.: Machine learning models for paraphrase identification and its applications on plagiarism detection. In: 2019 IEEE International Conference on Big Knowledge (ICBK), pp. 97–104 (2019)
6. Wang, B., Wang, A., Chen, F., Wang, Y., Kuo, C.-C. J.: Evaluating word embedding models: methods and experimental results. APSIPA Trans. Signal Inf. Process. **8**(1) (2019). https://doi.org/10.1017/ATSIP.2019.12
7. Tien, N.H., Le, N.M., Tomohiro, Y., Tatsuya, I.: Sentence modeling via multiple word embeddings and multi-level comparison for semantic textual similarity. Inf. Process. Manag. **56**(6), 102090 (2019). https://doi.org/10.1016/j.ipm.2019.102090
8. Dagan, I., Glickman, O., Magnini, B.: The PASCAL recognising textual entailment challenge. In: Quioñero-Candela et al. (ed.) First PASCAL Machine Learning Challenges Workshop, pp. 177–190 (2005)
9. Agirre, E., Cer, D., Diab, M., Gonzalez-Agirre, A., Guo, W.: SEM 2013 shared task: Semantic textual similarity. In: Second Joint Conference on Lexical and Computational Semantics (* SEM), vol. 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity, pp. 32–43 (2013)
10. Vulic, I., Moens, M.-F.: Cross-lingual semantic similarity of words as the similarity of their semantic word responses. In: Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT 2013), pp. 106–116 (2013)

11. Agirre, E. et al.: SemEval-2014 Task 10: Multilingual Semantic Textual Similarity. In: SemEval@ COLING, pp. 81–91 (2014)
12. Almarsoomi, F.A., OShea, J.D., Bandar, Z., Crockett, K.: AWSS: an algorithm for measuring Arabic word semantic similarity. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics, pp. 504–509 (2013)
13. Li, Y., Bandar, Z.A., McLean, D.: An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans. Knowl. Data Eng. **15**(4), 871–882 (2003)
14. Wali, W., Gargouri, B., Hamadou, A.B.: Sentence similarity computation based on WordNet and VerbNet. Comput. Sist. **21**(4), 627–635 (2017)
15. Schwab, D., Nagoudi, E.M.B.: Semantic similarity of arabic sentences with word embeddings. In: Third Arabic Natural Language Processing Workshop, pp. 18–24 (2017)
16. Nagoudi, E.M.B., Ferrero, J., Schwab, D., Cherroun, H.: Word embedding-based approaches for measuring semantic similarity of Arabic-English sentences. In: International Conference on Arabic Language Processing, pp. 19–33 (2017)
17. Ismail, S., Shishtawy, T.E., Alsammak, A.K.: A new alignment word-space approach for measuring semantic similarity for Arabic text. Int. J. Semantic Web Inf. Syst. IJSWIS **18**(1), 1–18 (2022)
18. Al-Theiabat, H., Al-Sadi, A.: The Inception Team at NSURL-2019 Task 8: Semantic Question Similarity in Arabic. ArXiv Prepr. ArXiv200411964 (2020)
19. Fadel, A., Tuffaha, I., Al-Ayyoub, M.: Tha3aroon at NSURL-2019 Task 8: Semantic Question Similarity in Arabic. ArXiv Prepr. ArXiv191212514 (2019)
20. Peters, M.E. et al.: Deep Contextualized Word Representations. arXiv preprint. ArXiv Prepr. ArXiv180205365 (2018)
21. Othman, N., Faiz, R., Smaïli, K.: Learning English and Arabic question similarity with Siamese neural networks in community question answering services. Data Knowl. Eng. **138**, 101962 (2022)
22. Daoud, M.: Novel approach towards Arabic question similarity detection. In: 2019 2nd International Conference on new Trends in Computing Sciences (ICTCS), pp. 1–6 (2019)
23. Boudaa, T., El Marouani, M., Enneya, N.: Alignment based approach for Arabic textual entailment. Procedia Comput. Sci. **148**, 246–255 (2019)
24. Boudaa, T., El Marouani, M., Enneya, N.: Arabic temporal expression tagging and normalization. In: International Conference on Big Data, Cloud and Applications, pp. 546–557 (2018)
25. Kuhn, H.W.: The Hungarian method for the assignment problem. Nav. Res. Logist. Q. **2**(1–2), 83–97 (1955)
26. Jonker, R., Volgenant, A.: A shortest augmenting path algorithm for dense and sparse linear assignment problems. Computing **38**(4), 4 (1987)
27. Drummond, O., Castanón, D.A., Bellovin, M.: Comparison of 2-D assignment algorithms for sparse, rectangular, floating point, cost matrices. In: Proceedings of the SDI Panels on Tracking, vol. 4, pp. 4–81 (1990)
28. ElKateb, S. et al.: Building a WordNet for Arabic. In: Proceedings of the 5th International Conference on Language Resources and Evaluation, Genoa, Italy, pp. 29–34 (2006)
29. Seelawi, H., Mustafa, A., Al-Bataineh, H., Farhan, W., Al-Natsheh, H.T.: Nsurl-2019 shared task 8: Semantic question similarity in Arabic. ArXiv Prepr. ArXiv190909691 (2019)
30. Pedregosa, F., et al.: Scikit-learn: machine learning in Python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
31. Altowayan, A.A., Tao, L.: Word embeddings for Arabic sentiment analysis. In: 2016 IEEE International Conference on Big Data (Big Data), pp. 3820–3825 (2016)

Evaluating Customer Segmentation Efficiency via Sentiment Analysis: An E-Commerce Case Study

Lahcen Abidar, Ikram El Asri, Dounia Zaidouni, and Abdeslam En-Nouaary

Abstract Understanding and efficiently utilizing client sentiment stands as a crucial pillar for achieving company excellence in today's constantly changing e-commerce landscape. We present a thorough framework for modeling sentiment analysis that is specially designed for the dynamic e-commerce market. We reveal a tremendous synergy that has the potential to completely change the industry by leveraging the capabilities of Natural Language Processing (NLP) in conjunction with customer feedback. By acknowledging how important customer reviews is in determining how successful e-commerce businesses are. With the aid of cutting-edge NLP techniques, we negotiate the complexities of sentiment analysis. Using the strength of NLP-enhanced customer evaluations, this study offers a solid foundation for modeling sentiment analysis within the e-commerce environment. It underscores the practical utility of sentiment analysis and its potential to drive positive transformations in the e-commerce industry.

Keywords Sentiment analysis · Nlp · Transformations · E-commerce

1 Introduction

In the ever-evolving landscape of e-commerce, the ability to understand and harness customer sentiment has transitioned from being merely advantageous to becoming an essential element for business success. With the proliferation of online shopping and

L. Abidar (✉) · I. El Asri · D. Zaidouni · A. En-Nouaary
Mathematics and Computer Science, INPT, Madinat Al Irfane, Rabat 100190, Rabat, Morocco
e-mail: abidar.lahcen@inpt.ac.ma

I. El Asri
e-mail: elasri@inpt.ac.ma

D. Zaidouni
e-mail: zaidouni@inpt.ac.ma

A. En-Nouaary
e-mail: abdeslam@inpt.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_18

the digitalization of consumer experiences, e-commerce businesses are increasingly reliant on customer feedback and reviews to shape their strategies and offerings [1, 2].

This paper is dedicated to evaluating the efficiency of a customer segmentation module [3–5] in the context of e-commerce. We present a comprehensive framework for sentiment analysis meticulously crafted to suit the dynamic nature of the e-commerce market and its role in customer segmentation. In a world where customer opinions are shared and scrutinized on digital platforms, the importance of sentiment analysis within the context of customer segmentation cannot be overstated. Moreover, the advent of Natural Language Processing (NLP) has opened new avenues for extracting rich insights from customer feedback [6, 7]. Recognizing that customer reviews wield considerable influence over the fortunes of e-commerce enterprises, this study endeavors to decode the intricacies of sentiment analysis and its role in enhancing customer segmentation. We delve into the world of NLP to extract valuable insights from customer evaluations, extending beyond mere sentiment polarity classification to uncover the nuanced subtleties and intricate patterns inherent in e-commerce customer sentiments [8, 9]. This research embarks on a comprehensive journey through a diverse array of e-commerce reviews, encompassing product assessments, service appraisals, and customer experiences. The methodology employed hinges on data-driven decision-making, enabling businesses to derive actionable insights. These insights, in turn, guide strategic adjustments aimed at enhancing customer satisfaction, operational efficiency, and the overall performance of the customer segmentation module [10, 11]. Furthermore, this paper introduces an innovative strategy for targeted action, a forward-looking approach nurtured by the insights gained through sentiment analysis and the evaluation of the customer segmentation module. This strategy emphasizes not only the practical applicability of sentiment analysis but also its potential to reshape the e-commerce industry through more effective customer segmentation. By facilitating personalized customer experiences, fine-tuning operations, and adapting to evolving market dynamics, sentiment analysis emerges as a catalyst for transformative change within the e-commerce sector, especially when integrated with advanced customer segmentation techniques [12, 13].

The remainder of this paper is organized as follows. A brief overview of the most recent initiatives regarding the field of sentiment analysis in the context of e-commerce is provided in Sect. 2. Section 3 presents a Workflow of our proposed model. In Sect. 4, we describe an empirical case study. Section 5 discusses the results. Finally, Sect. 6 concludes the paper and highlights some future directions.

2 Related Work

The field of sentiment analysis in the context of e-commerce has garnered substantial attention from researchers and practitioners alike. In this section, we provide an overview of relevant research and approaches that have contributed to our understanding of sentiment analysis in the e-commerce domain. Researchers have focused

on leveraging customer reviews and feedback to assess the sentiment associated with products, services, and overall shopping experiences. Early studies primarily employed rule-based methods and lexicon-based approaches to classify sentiments as positive, negative, or neutral [14]. Recent advancements in machine learning and Natural Language Processing (NLP) techniques have propelled sentiment analysis to new heights. Deep learning models, such as recurrent neural networks (RNNs) and convolutional neural networks (CNNs), have demonstrated superior performance in capturing nuanced sentiment patterns in customer reviews [15, 16]. The integration of NLP techniques has significantly enhanced the capabilities of sentiment analysis in e-commerce. Researchers have harnessed NLP for tasks such as aspect-based sentiment analysis, where sentiments are attributed to specific product features or aspects [17]. Additionally, techniques like sentiment lexicon expansion and sentiment lexicon adaptation have improved sentiment classification accuracy [18, 19]. Sentiment analysis findings have not only served as a basis for understanding customer preferences but have also played a pivotal role in generating personalized product recommendations. Recommender systems that incorporate sentiment analysis results have proven to be more effective in tailoring product suggestions to individual users, thereby enhancing user satisfaction and boosting sales [20].

As sentiment analysis continues to influence e-commerce decision-making, ethical considerations have come to the forefront. Concerns related to privacy, bias, and transparency in sentiment analysis algorithms have spurred discussions on responsible AI deployment in e-commerce [21, 22]. The impact of sentiment analysis extends beyond academic research, with numerous e-commerce companies implementing sentiment analysis tools and platforms. These tools aid in reputation management, customer support, and product improvement efforts [23].

Our research paper centers on the analysis of customer reviews pertaining to customer segmentation to assess whether the segmentation strategies implemented within an organization's customer base elicit predominantly positive or negative feedback. The findings derived from this analysis are one of the main evaluations of these strategies.

3 Workflow Model

In this section, we introduce the framework for processing and analyzing customer reviews. The workflow model is represented visually in Fig. 1, which provides an overview of the entire process.

3.1 Data Preprocessing

Handling Missing Values: Two common approaches for dealing with missing data in operational datasets are: (1) discarding samples with missing values, suitable when

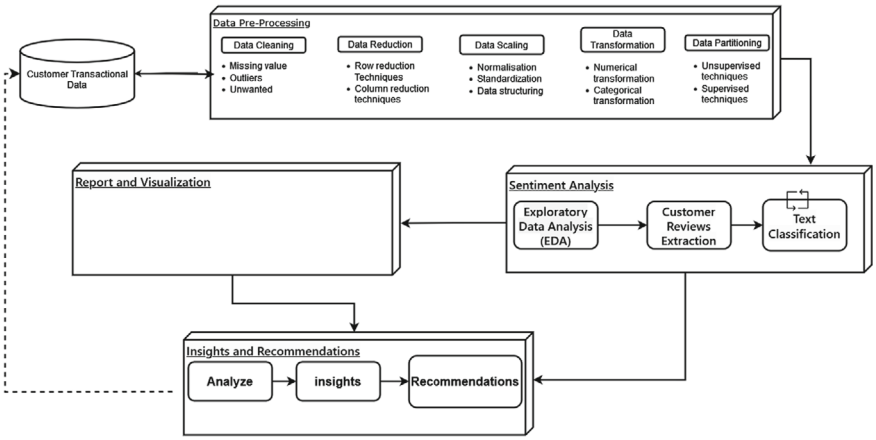


Fig. 1 Workflow

missing values are minimal, and (2) applying missing value imputation methods to replace missing data with inferred values [24].

Data Reduction: Data reduction includes both row-wise (sample reduction) and column-wise (variable reduction) techniques. Row-wise reduction can be achieved using methods like random and stratified sampling. Feature extraction, on the other hand, constructs new features from existing variables through linear or nonlinear combinations [24].

Data Scaling: Data scaling ensures predictive modeling validity, particularly when input variables have varying scales. Two common scaling methods are max-min normalization ($x' = (x - x_{\min})/(x_{\max} - x_{\min})$) and z-score standardization ($x' = (x - \mu)/\sigma$), where $x_{\min}$ and $x_{\max}$ represent the variable's minimum and maximum values, μ is the mean, and σ is the standard deviation [24].

Data Transformation: In the building field, data transformation often converts numerical data into categorical data to align with data mining algorithms. Common methods include equal-width and equal-frequency transformations for simplicity [24].

Data Partitioning: Data partitioning divides the dataset into groups for in-depth analysis. Widely used techniques in the building field include clustering analysis (e.g., k-means, hierarchical clustering) and decision tree methods [24].

3.2 Sentiment Analysis

Exploratory Data Analysis (EDA): In the EDA section, we conduct a preliminary examination of the dataset to gain insights into customer reviews and their characteristics. This includes: Data Overview,Descriptive Statistics, Distribution Analysis, Text Length Analysis, Word Clouds.

Customer Reviews Extraction: This section is dedicated to the procedure of collecting and preparing textual data from the dataset to facilitate sentiment analysis. This entails three key steps: Data Preprocessing, Feature Extraction, and Text Tokenization.
Within the realm of customer reviews, Natural Language Processing (NLP) stands as an indispensable tool. It empowers us to immerse ourselves in the extensive pool of information embedded in these textual reviews and derive valuable insights. This NLP-driven methodology begins with data understanding, enabling us to gain a deeper understanding of customer feedback. Subsequently, we employ regular expressions to address various aspects of the text, including line breaks, hyperlinks, dates, monetary values, numbers, negations, special characters, and redundant whitespaces. In addition, we eliminate stopwords, which are frequently occurring words that contribute limited value to the analysis. The process culminates in stemming, a technique that simplifies words to their root forms, rendering the text more amenable to further analysis. Leveraging the capabilities of NLP empowers us to unlock the wealth of information contained within customer reviews, enabling data-driven decisions and the enhancement of overall customer satisfaction.

Text Classification: In the "Text Classification", the core of sentiment analysis is covered, including, Model Selection, Training and Evaluation, Results Interpretation, Visualization of Results.

3.3 Report and Visualization

In the "Report and Visualization" section, a succinct summary of the principal findings derived from the sentiment analysis is presented, accompanied by the incorporation of visual aids such as charts and graphs. These visual representations serve to effectively depict the distribution of sentiments and the trends observed within the customer reviews.

3.4 Insights and Recommendations

In the "Insights And Recommendations" section, we delve into a comprehensive analysis of the sentiment analysis results, aiming to provide a deep understanding of the findings.

Analyze: In this subsection, we explore the distribution of sentiment labels, investigate the impact of various features on sentiment, and analyze sentiment variations across product categories. We also examine temporal trends, compare results with industry benchmarks, and perform qualitative analysis on select reviews. This section provides a comprehensive analysis of sentiment results.

Insights: Summarizing the key insights derived from our sentiment analysis, we highlight trends in customer sentiment, product performance, and customer expectations. We discuss seasonal variations and their implications, along with competitive analysis and the impact of marketing initiatives. These insights serve as a foundation for data-driven decision-making.

Recommendations: In the recommendations section, we provide actionable steps to improve various aspects of the e-commerce ecosystem. These recommendations are based on the insights gained from sentiment analysis and aim to enhance customer satisfaction and overall business performance. Recommendations include product enhancements, customer service improvements, marketing strategy adjustments, and a focus on product categories with positive sentiment. We also emphasize the importance of a feedback loop with customers and data-driven decision-making.

4 Empirical Results and Analysis

in this section, we will train our sentiment analysis model with customer reviews. To do this, we will need labeled data where each review is associated with a sentiment label (e.g., positive, negative).

4.1 Data

In this use case, we will use The Olist Store dataset [25] is a publicly available dataset from Brazilian e-commerce that comprises information on 100,000 orders placed between 2016 and 2018 across various Brazilian marketplaces. This dataset

offers a multifaceted view of each order, covering aspects such as order status, pricing, payment, shipping performance, customer location, product attributes, and customer reviews.

4.2 Natural Language Processing

Table 1 illustrates comprehensive representation of the sentiment analysis process on customer comments. It shows the transformation of raw text data through regular expressions and preprocessing to the final sentiment classification. Each processing step, such as retaining line breaks or removing stopwords, is shown in separate columns, making it clear how the text is transformed at each stage of analysis. The sentiment label and sentiment score provide insights into the overall sentiment expressed in the customer reviews, whether positive or negative, and the degree of sentiment polarity, represented as a percentage (PCT).

4.3 Feature Engineering

Convert the text data into numerical representations, such as word embeddings or TF-IDF (Term Frequency-Inverse Document Frequency) vectors. These numerical features are used as input to machine learning models.

$$TF = \frac{Frequency|of|word|in|the|document}{Total|words|in|the|document} \tag{1}$$

$$IDF = \frac{Total|number|of|docs}{Number|of|docs|containing|the|words} \tag{2}$$

4.4 Data Labeling

Labeling our reviews is typically done through manual annotation, where we assign sentiment labels. However, due to the dataset's extensive size and resource limitations that prevent us from outsourcing this task, we face a challenge. The dataset lacks explicit labels indicating comment sentiment (positive or negative) (Table 1). To address this, we propose an alternative approach: manually examining comments and assigning labels as 1 for positive and 0 for negative. While this method is more accurate, it's time-consuming. For a more efficient implementation, we'll use the review score column, categorizing comments with scores 1, 2, and 3 as negative and those with scores 4 and 5 as positive.

Table 1 NLP pipeline

Order_id	Score	Comment	Re_breakline	Re_hiperlinks	Re_dates	Re_money	Re_numbers	Re_negation
3	4	Mas um pouco, travando...pelo valor ta Boa	Mas um pouco, travando...pelo valor ta Boa	Mas um pouco, travando...pelo valor ta Boa	Mas um pouco, travando...pelo valor ta Boa	Mas um pouco, travando...pelo valor ta Boa	Mas um pouco, travando...pelo valor ta Boa	Mas um pouco, travando...pelo valor ta Boa
17	1	Nada de chegar o meu pedido	Nada de chegar o meu pedido	Nada de chegar o meu pedido	Nada de chegar o meu pedido	Nada de chegar o meu pedido	Nada de chegar o meu pedido	Nada de chegar o meu pedido
Order_id	Score	Re_special_chars	Re_whitespaces	Stopwords_removed	Stemming	Sentiment_label	Sentiment PN	PCT
3	4	Mas um pouco travando pelo valor ta Boa	Mas um pouco travando pelo valor ta Boa	pouco travando valor ta boa	pouc trav val ta boa	positive	Positive	98
17	1	Nada de chegar o meu pedido	Nada de chegar o meu pedido	nada chegar pedido	nad cheg ped	negative	Negative	−64

4.5 Model Selection

By utilizing Logistic Regression, we aim to create a predictive model that effectively distinguishes between positive and negative sentiment in customer reviews. This model will be a critical component of our sentiment analysis pipeline, contributing to the automated labeling and classification of comments.

4.6 Model Training

we split the dataset into training and testing subsets, with 80% of the data allocated for training and the remaining 20% reserved for testing. This division is crucial to assess the model's performance, validate its effectiveness, and ensure it generalizes well to unseen data. The random state is set to 42 for reproducibility and consistency in our analyses 1.

4.7 Evaluation

In the training phase, our model achieved an accuracy of 0.8865, demonstrating its ability to correctly classify sentiments. Precision, recall, F1-score, and the area under the curve (AUC) were all indicative of the model's strong performance, with precision at 0.9226, recall at 0.9227, F1-score at 0.9226, and AUC at 0.9436. The accuracy for the testing phase was 0.8857, indicating that the model effectively generalizes to new, unseen data. Precision, recall, F1-score, and AUC remained strong, with precision at 0.9248, recall at 0.9198, F1-score at 0.9223, and AUC at 0.9447.

5 Result and Discussion

The results of our analysis reveal a total count of reviews classified into two sentiment categories: Negative and Positive. The Negative category comprises 10,416 reviews, while the Positive category includes 30,561 reviews, as illustrated in Fig. 2. These figures represent the outcomes of a targeted campaign initiated using our customer segmentation model to classify and address specific customer segments. The primary objective of this analysis is to assess the efficiency of our segmentation model by evaluating the distribution of sentiments within these reviews. By contrasting the total counts of Negative and Positive reviews, our aim is to glean valuable insights into the effectiveness of our segmentation approach and its impact on customer sentiments. These findings shed light on the success of our strategy in catering to the unique needs and preferences of different customer segments, thereby enhancing overall

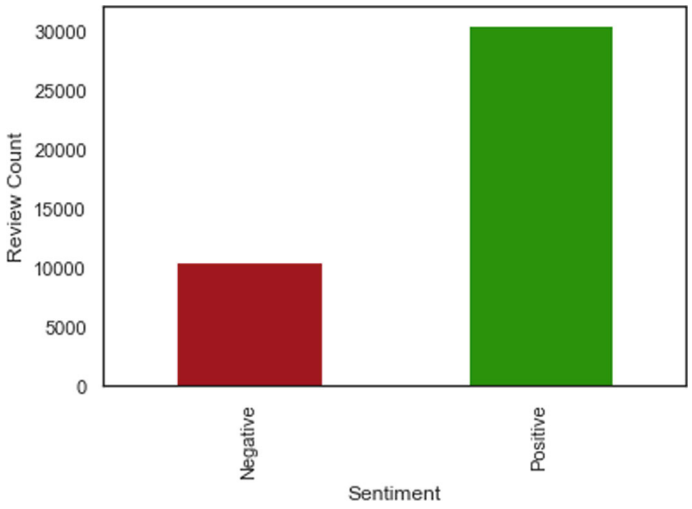


Fig. 2 Sentiment distribution

customer satisfaction and loyalty. Such insights are pivotal for refining our business strategies and customer engagement approaches. Leveraging sentiment analysis on customer reviews, our model swiftly assesses the prevailing sentiments-negative and positive-within our customer base. By deriving a percentage of positive and negative sentiments, we gain immediate insights into the reception of our segmentation strategies. This approach allows us to dynamically gauge the impact of our segmentation efforts without the need to wait for the culmination of a quarter or monthly. This agile methodology not only accelerates our decision-making process but also empowers us to make timely adjustments to our segmentation strategies, ensuring that we are responsive to customer feedback and continuously refining our approach for optimal customer satisfaction.

6 Conclusion and Future Directions

In this paper, we embarked on a comprehensive journey into the world of sentiment analysis and its application to customer reviews, a vital component of our customer segmentation efforts. Through the use of the e-commerce dataset, we gained valuable insights into customer sentiments, enabling us to distinguish between positive and negative reviews, a fundamental aspect of assessing the efficiency of customer segmentation. As we look ahead with a focus on customer segmentation efficiency, there are several promising directions for further research and development. It is crucial to conduct extensive longitudinal studies to assess the long-term impact of customer

segmentation and sentiment analysis on customer loyalty and overall company revenue. By implementing the model across multiple business cycles, we can gain a more comprehensive understanding of its effectiveness and its potential contribution to revenue growth, thereby validating the sustainability and adaptability of our segmentation strategies. Furthermore, future research can delve into more advanced sentiment analysis techniques, including natural language understanding and sentiment evolution modeling. These advanced approaches will enable us to gain deeper insights into customer experiences and evolving sentiment trends, equipping us to make real-time adjustments to our segmentation strategies.

References

1. Smith, J.: Understanding customer sentiment in e-commerce. J. E-Commerce Res. (2022)
2. Chen, A.: E-commerce trends: a comprehensive analysis. Int. J. E-Commerce (2021)
3. Abidar, L., Zaidouni, D., Ennouaary, A.: Customer segmentation with machine learning: new strategy for targeted actions, pp. 1–6 (2020)
4. Abidar, L., Asri, I.E., Zaidouni, D., Ennouaary, A.: A data mining system for enhancing profit growth based on RFM and CLV. In: 2022 9th International Conference on Future Internet of Things and Cloud (FiCloud), pp. 247–253 (2022)
5. Abidar, L., Zaidouni, D., Asri, I.E., Ennouaary, A.: Predicting customer segment changes to enhance customer retention: a case study for online retail using machine learning. Int. J. Adv. Comput. Sci. Appl. **14** (2023)
6. Brown, D.: Advancements in natural language processing. NLP Adv. (2020)
7. Liu, W.: NLP techniques for text analysis. Text Mining J. (2019)
8. Gupta, R.: Unlocking the Secrets of Sentiment Analysis. Sentiment Insights (2021)
9. Goldberg, S.: Sentiment analysis beyond polarity. Text Anal. J. (2017)
10. Wang, L.: Data-driven decision-making in e-commerce. E-Commerce Anal. (2020)
11. Rajapakse, N.: Improving operational efficiency through data analysis. Oper. Manage. Rev. (2018)
12. Duan, M.: Innovative strategies in e-commerce. E-Commerce Strategy J. (2019)
13. Verbeke, C.: Adapting to market dynamics: the e-commerce challenge. Market Dyn. Rev. (2021)
14. Liu, B.: Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies (2012)
15. Kim, Y.: Convolutional Neural Networks for Sentence Classification, pp. 1746–1751 (2014)
16. Tang, D., Qin, B., Liu, T.: Document modeling with gated recurrent neural network for sentiment classification. In: Proceedings of the 2015 Conference on Empirical Methods in (EMNLP), pp. 1422–1432 (2015)
17. Schouten, K., Frasincar, F., de Jong, F.: Aspect-based sentiment analysis with gated recurrent nn. Exp. Syst. Appl. **152**, 113351 (2020)
18. Ma, D., Li, S., Zhang, X., Wang, H.: Interactive attention networks for aspect-level sentiment classification. In: Proceedings of the 56th Annual Meeting of the (ACL), pp. 646–655 (2018)
19. Khan, M.S., Lin, C., Tao, X.: Aspect-based sentiment analysis: a comparative analysis and survey. ACM Comput. Surv. (CSUR) **52**, 92:1–92:34 (2019)
20. Zhao, J., Cheng, H., Ma, S., Zhang, Y.: Personalized recommendation based on sentiment analysis for e-commerce platforms. Appl. Soft Comput. **104**, 107283 (2021)
21. Hajian, S., Rudzicz, F.: Ethical considerations in AI research: a case study of sentiment analysis in mental health. In: Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency (FAT* '20), pp. 607–617 (2020)

22. Mehrabi, N.L.S., Morstatter, F., Peng, N., Galstyan, A., Lerman, K.: A survey on bias and fairness in machine learning (2019). arXiv preprint arXiv:1908.09635
23. Rahmani, R., Armand, M., Falahollahi, M., Kahani, M., Crestani, F.: An overview of sentiment analysis in e-commerce. Inform. Process. Manage. **58**, 102513 (2021)
24. Fan, C., Chen, M., Wang, X., Wang, J., Huang, B.: A review on data preprocessing techniques toward efficient and reliable knowledge discovery from building operational data. Front. Energy Res. **9** (2021)
25. olist. Olist store dataset. https://www.kaggle.com/olistbr/brazilian-ecommerce (2023)

Enhancing Sentiment Analysis in Moroccan Mixed Script: A Case Study of Perspectives on Distance Learning During the Covid-19 Pandemic

Monir Dahbi, Samir Mbarki and Rachid Saadane

Abstract Sentiment analysis in Arabic texts faces challenges with mixed scripts, including foreign words, Arabizi, and dialectal variations. This paper presents a methodology for standardizing mixed-script texts, employing translation and transliteration techniques. We have implemented a preliminary step to translate foreign words before delving into Arabizi. Thereafter, a set of rules to facilitate the transition from Arabizi to Arabic. Through this study, we try to discover the behaviour of Moroccan Internet users towards the distance education system. The remote teaching process has faced a lot of criticism, such as the lack of equal opportunities among pupils; we find that the parents refuse to pay for the remote studies of their children during the quarantine period. On the other hand, the supporters say that the new system contributed to the continuation of the educational process, saving time and traditional teaching expenses, and look at it as a good opportunity to raise a generation that is experienced in digital technology. To address the challenges mentioned earlier, this paper presents an improved methodology for analysing Moroccan data on the social networking platform, analysing a group of opinions on these topics by extracting 4794 tweets detailing the process of pre-processing Moroccan text and comparing several machine-learning algorithms (Naïve Bayes and Support Vector Machine). Experimental results have achieved good accuracies and confirmed the effectiveness of the proposed approach.

Keywords Arabizi · Morocco · Machine learning · Sentiment analysis

M. Dahbi (✉) · S. Mbarki
Department Computer Science, Ibn Tofail University, Kenitra, Morocco
e-mail: monir.dahbi@gmail.com

R. Saadane
Hassania School of Public Work (EHTP), Casablanca, Morocco

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_19

1 Introduction

The pandemic has changed people's daily habits and activities. Education and working from home have become a necessity. The teacher strives to teach his students. Since the outbreak of the Coronavirus in Morocco, the Moroccan Ministry of National Education and Higher Education has resorted to strategic approaches to enable students in different cities and villages to receive their education, even remotely. Some cultural TV channels have become a ready-to-learn platform. Moreover, the teachers had the opportunity to excel in teaching their students. To evaluate people's opinions about the effectiveness of this procedure, as well as to monitor the problems that some actors and recipients suffer from this obligatory change during this period, we will focus in this research on the following topics:

– Distance education performance evaluation.
– Problems with the payment of dues for private schools in the eyes of parents.
– Problems with the payment of dues for private schools in the eyes of teachers.

Twitter is one of the most popular social networks. So as, to better understand and categorize tweets circulating between Internet users, the field of automatic natural language processing (NLP) has been adopted to answer this necessity. The simplest definition of (NLP) is the task of processing and analysing natural (spoken) languages that are written, to implement an optimal interaction between humans and computers. Sentiment analysis is one of the most important areas of Natural Language Processing (NLP); defined as the process of extracting useful models from textual data. These useful models include interpreting and classifying the sentiment as neutral, positive, or negative from the collected data using specific analysis techniques [1]. From a business perspective, by analysing users' opinions and interactions, such as searching for posts or comments about a product on social media platforms, which will enable companies to respond to their needs and improve their services and products. Almost all research done on sentiment analysis was about English texts. Only a few worked on Arabic texts and dialects. The dialect does not contain clear grammatical rules and contains many different ways of writing, which makes the analysis process more difficult. The purpose of this paper. First, is to present a new framework for the Moroccan sentiment analysis, using text classification applied to different datasets related to the previously mentioned topics. The particularity of this part is facing the challenge of processing a mixed script: (MSA) alongside the Moroccan dialect and other languages. The second goal is the use of a few different algorithms to find those that provide the best results in terms of sentiment classification. Finally, to demonstrate how our framework can be a useful tool for understanding people's reactions. What distinguishes this research is the use of an improved approach for the analysis of Moroccan sentiments by adopting a hybrid method for text processing.

The proposal consists of several sections. The second paragraph discusses related work on sentiment analysis from social media interactions. The overall

plan of the proposed approach is outlined in the third paragraph. The fourth paragraph covers experiments and the obtained results. Conclusions and future work are presented in the fifth.

2 Related Work

A lot of research done on sentiment analysis focused on analysing Twitter posts, most of them were written in English and Arabic. With little research on the dialect. Here is a set of methods used by some researchers:

ALSALEEM, Saleh, et al. took the subject of automatic classification of text documents in Arabic and applied the Naïve Bayesian algorithm (NB) and Support Vector Machine algorithm (SVM) to these data sets. The results show that the SVM algorithm gave better results than the NB algorithm [2].

Researchers DoniaGamal, Marco Alfonse, et al. Reported on the difficulties that remain an obstacle to the classification of sentiments in Arabic, especially for dialects (Egyptian as an example). They proceeded with these challenges by cleaning data and then chose to use supervised machine learning algorithms (SVM, NB, SGD, MNB, BNB, and LR) to identify the sentiments of tweets. The experiments made confirmed that the SVM classifier has the highest accuracy of 93.57% [3].

To address the opinions of Saudis during the period of the coronavirus, ADDAWOOD, Aseel et al. made the extraction of data available to the public and then constructed a lexicon of 7,534 words. The accuracy of classifiers was good, the SVM classifier was better than NB with 98%, while the accuracy of NB was 87%. The results showed that people had a positive reaction during this period [4].

To analyse the ABSA of Arab hotel reviews. Al-Smadi, et al. present some methods based on supervised machine learning (deep RNN and SVM) and trained with lexical, verbal, syntactic, morphological, and semantic characteristics. The evaluation shows that the SVM algorithm gave better results than deep RNN on analysed data [5].

Stemming Arabic terms has proven in several pieces of research that it is not an easy task because of its highly inflected and derivational nature [6]. Proven that light stemming allows remarkably good information retrieval without providing correct morphological analyses. Developed several light stemmers for Arabic, and assessed their effectiveness for information retrieval using standard TREC data. Khoja and Garside developed and evaluated a new Arabic stemmer, which reduces the words to their roots. Their stemmer removes all the punctuation marks, diacritics, numbers, the article "ال", "the", and the inseparable conjunction prefixes "و", "and". Additionally, they have built a large prefixes' and suffices' list, which is used to check all the input words if they include any of them and the longest of these stripped off if found. Finally, the produced word is then, compared against a list of patterns and if a match is found, the root is produced [7].

3 Proposed Approach

Nowadays, people use more and more microblogging platforms like Facebook and Twitter, which generate an enormous amount of data, so the analysis of this data will undoubtedly help to understand peoples' opinions and contribute to decision-making, in terms of monitoring citizens' needs and making plans [8]. The objective of this work is to create a framework for monitoring sentiment on social networks to extract people's opinions on the distance learning system, based on the extracted tweets.

To achieve our goals, our method divided into four major processes: Data collection and annotation, Data pre-processing, Opinion classification, and Results validation.

Figure 1 shows the main methodology. The first module consists of data collection and annotation. The second module consists of the pre-processing of the corpus to clean and correct the text. The third module involves the detection of aspects by using machine-learning classification techniques. The last step allows the calculation of the efficiency of each algorithm.

3.1 Data Collection and Annotation

Moroccan Arabic subdivided into two parts:

- Modern Standard Arabic (MSA), which represents the official language of the country and is widely used in official, cultural and informative speeches as well as in education.
- The Moroccan dialect (MD) is the native language of many Moroccans with much overlap with standard Arabic.

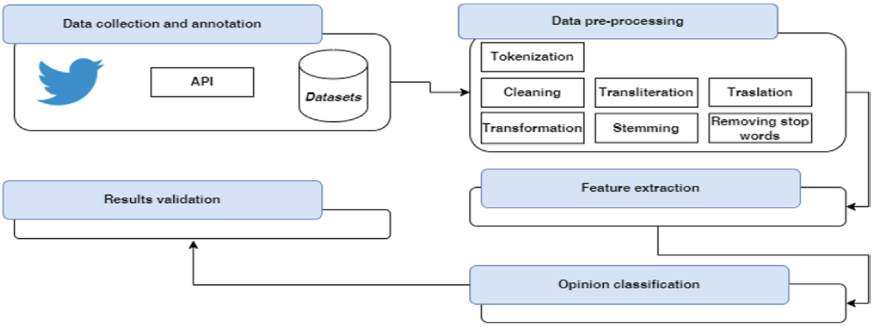


Fig. 1 Proposed methodology

Technological developments, such as cell phones and the advent of the Internet, have led to the emergence of a new and unique form of writing, widely used in chat. This new style of writing combines Latin letters and numbers. Known as Arabish or Arabizi [9]. We collected data from the Twitter platform during the period from 03/16/2020 to 07/31/2020, the period during which the epidemic first spread to Morocco, where the government decided to close schools and universities to prevent the spread of the pandemic. We collect data using a Python API named Tweepy [10]. Tweepy is an API allowing a Twitter user to access data via coding, by creating a Twitter application, while retrieving the Twitter Developer API key to be able to authenticate and access the data [11]. The tweets were extracted using keywords related to the distance education system, as well as the reaction of Moroccans to private education in this period. Table 1 shows examples of extracted tweets about distance learning. Table 2 shows the number of extracted tweets per each topic.

We made a manual annotation of the processed data. It is either positive, negative or neutral.

Table 1 Example of extracted tweets

	Tweet	English translation	Sentiment
Topic 1	#التعليم عن بعد Afchal 9arar !	Distance education #failed decision!	Negative
	Mzian lina nstafdo b technologie pour assurer la continuité dial taalim. #التعليم_عن_بعد #المغرب_	We will use technology to ensure the continuity of educqtion. #Distance_Education #Morocco	Positive
Topic 2	للأسف، المشاكل مع سداد فلوس المدارس الخاصة كتكبر وقت كورونا. الآباء والأمهات	Unfortunately, the problems with paying private school funds have grown during the time of Corona. fathers and mothers	Negative
	بالتضامن والتعاون، غادي نجيبو على هاد التحدي بفعالية #المغرب #كورونا #مدارس_خاصة	With solidarity and cooperation, we will meet this challenge effectively #Private_schools #Corona #Morocco	Positive
Topic 3	المدارس الخاصة فالمغرب كتعيش لحظات صعبة ومعاناة بزاف وقت كورونا. الأباء ماكايفهموش الضغط المالي للمدارس. #المدارس_الخاصة #كورونا #المغرب	Private schools in Morocco are experiencing difficult moments and great suffering during the time of Corona. Parents do not understand the financial pressure of schools. #Private_Schools #Corona #Morocco	Negative
	غانحتاجوا دعم وفهم أكبر حتى يستمر التعليم بكفاءة #المدارس_الخاصة #كورونا #المغرب	They need greater support and understanding so that education can continue efficiently #Private_ Schools #Corona #Morocco	Positive

Table 2 The number of extracted tweets per topic

Distance education performance	1690
Problems with the payment of dues for private schools for parents	1570
Problems with the payment of dues for private schools for teachers	1534

3.2 Data Pre-Processing and Feature Selection

Pre-processing techniques are a crucial step in SA for Arabic text. Especially Arabic dialect text due to its unstructured form.

As mentioned in (Fig. 1), the pre-processing module involves seven processes:

Tokenization

Tokenization is the segmentation of continuous text into discrete units, facilitating analysis and processing in natural language processing tasks.

Cleaning

This part is essential and serves to eliminate all impurities: Additional white space; URLs, which start with http:// or https://; Twitter user's name like @user_name; punctuation.

Transformation

At this stage, we use a set of dictionaries, such that abbreviations, emoticons and acronyms for:

- Transforming every acronym name and emoticon in their expression mapped into its corresponding word.
- Removing repetitions (or exaggeration) of alphabets that appear in a word more than once. For example, زووووين transformed into زوين
- Removing Attatweel (The lengthening) of the character '_' بحـــــال transformed into بحال
- Removing Attashkeel (supplementary diacritics that given to the alphabets) for example بُزَّاف transformed into بزاف

Translation

During this part of treatment, a translation mill is used to detect the language of the word and translate each word can be translated into Modern Standard Arabic MSA that recognized as foreign. We have used a Python script that relies on Google API. Figure 2 shows the general architecture of translation and transliteration.

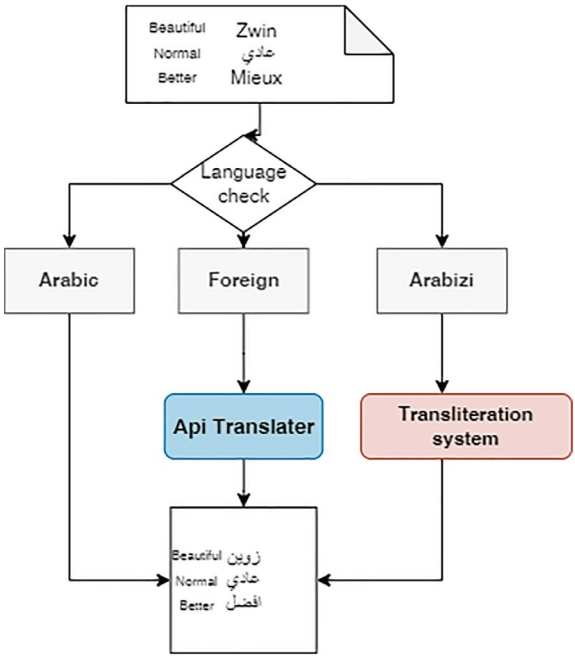


Fig. 2 General architecture of translation and transliteration

Transliteration

The approach used contains three phases. (1) Extraction of the Arabic/Arabizi corpus. (2) Generation of candidates. (3) Selection of the best candidate. We present in (Fig. 3) the general architecture of the proposed approach. The details of each phase will detailed in the following.

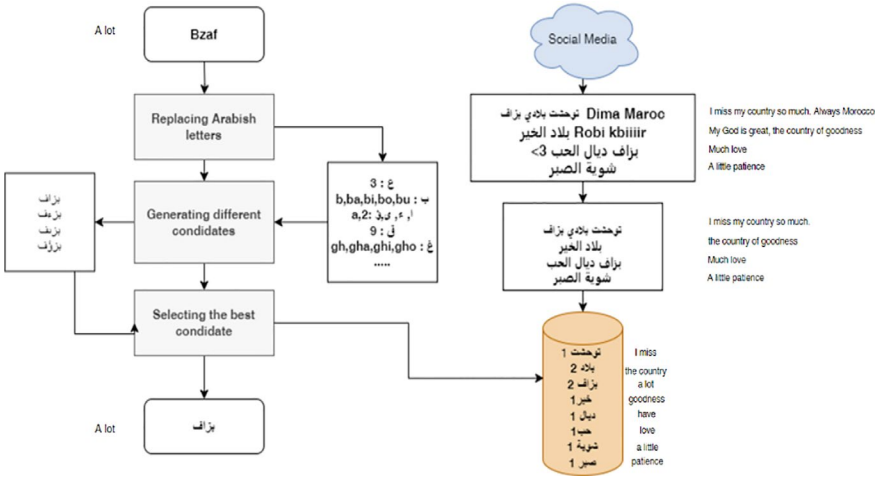


Fig. 3 General architecture of Arabizi transliteration

We start by automatically extracting an Arabic/Arabizi corpus from messages posted by Moroccan internet users on social media. A corpus containing 17,984 messages is collected. After filtering out non-Arabic messages, 11,232 messages retained. Our focus is solely on messages written in Arabic characters. To ensure the quality of the corpus a set of pre-processing methods are used: (1) delete repeated messages; (2) delete exaggerations (for example the word "Zwiiiiiin" is transformed into "zwin": and "bzaaaf" is transformed into "bzaf"; (3) delete the character '# ' and different punctuation '.,!,? '; (4) delete consecutive whites spaces as well as Tatweel ('–'). We finish by dividing our corpus into a bag of words, where each word listed along with a number symbolizing the frequency of its repetition.

By Applying the different replacements (based on the different possibilities) presented in Table 3, each Arabizi word generates several words in Arabic.

To identify the most accurate transliteration of a given Arabizi word into Arabic, we use a basic linguistic model. The construction of this model is based on a large Arabic corpus from social media, which is subject to pre-processing. The linguistic model encompasses the distinct words found in the corpus as well as their respective frequencies. The approach is to search for each candidate within this model, extract the frequency of each word and finally select the candidate with the highest frequency.

Table 3 Examples of Arabizi letters and their correspondents in Arabic

Arabic characters	Characters in Arabish	Arabic characters	Characters in Arabish
ث ص س	s, sa,si,so	أ إ ء آ ئ ؤ	2
ش	*sh sha,shi sho*	ع	3
ب	p, po,pi,pa,pu	أ ا ة ى ع	a
ط ت	t, ta,ti,to,tu	خ	5
و	w, wa,wi,wu,wo	ط	6
ج	j, g	ح	7
ى ي	y	ق	9
ي	ya yi yo yu	ع	3a,3i,3e,3o,3u
ز ظ ذ	z, za,zi,zo,zu	ب	b, ba,bi,bo,bu
م	m, ma,mo,mi	ش	ch,cha,chi
ن	n	ض د	d, da,di,do
ذ ظ	dh	غ	gh, gha, ghi, gho
ج	dj, dja,dji, djo	ه	h, ha,hi,ho
ف	f, fa,fi,fo	ي	i, ia,ie,ii,io,iu
ق	9a,9o, 9 e,9i	خ	kh, kha,khi,kho
ك	k, ka,ki,ko,ke	و	o,oa,oi,ou,oo
ل	bi	و ا	o

Table 4 The candidates generated from the Arabizi transliteration

"bzaf"		"chwya"	
Candidate	Frequency	Candidate	Frequency
بزأف	18	شوياً	0
بزاف	546	شويا	22
بزتف	0	شوية	398
بزبف	0	شوبى	0
بزعف	28	شويع	0
بذأف	0	شوياً	0
بذاف	0	شويا	192
بذبف	0		
بذعف	0		
بظأف	0		
بظاف	0		
بظتف	0		
بظعف	0		

In the provided example, specifically for the words "bzaf" and "chwya," the optimal candidates are "بزاف" and "شوية." A significant number of other generated candidates yield a probability of 0 as they lack meaningful equivalents in Arabic and its dialects. Some alternative candidates may have non-zero probabilities, but these are generally insignificant compared to the probability of the best candidate. In instances where no candidate is found in the model, the best candidate is determined using the rules associated with the initial letter, as outlined in Table 4.

Removing stop words

Stop words act as neutral words that do not influence the polarity of the sentence. They are usually filtered during the pre-processing of the text, but in our case, we do not initially deal with a hundred per cent Arabic words (MSA), so this process takes place after the standardization of writing. The filtering of stop words decreases the number of words we do sentiment analysis for, which reduces space and time. Here are some examples of Arabic stop words: سوف، كيف، حتى. In our case, we included a list of stop words based on research conducted by (Khoja and Garside) [7] who compiled a list of 168 stop words that have been used in the research done by (Larkey and Connell) [13].

Stemming

After the translation phase, we have relatively only words written in Arabic letters. As mentioned in paragraph 2, we will implement an Arabic ISRI Stemmer from the NLTK libraries, [12], to be able to reduce as much as possible variant word forms related to the common roots into a single word. Table 5 show examples of sequence of steps.

Table 5 Examples of sequence of steps

Step	Text	English text
Original text	التحية للتعليم عن بعد فالمغرررب ! Technologies 3awnatni bzaaaf كيخدمو مزيان #Corona	Salute to distance learning in Morocco! Technologies 3awnatni bzaaaf they work well #Corona
Tokenization	التحية / للتعليم / عن / بعد / فالمغرررب 3awnatni / Technologies / ! #Corona / مزيان / كيخدمو / bzaaaf	in Morocco / distance learning / Greeting to ! / helped me a lot / Technologies #Corona / They work well
Cleaning	التحية / للتعليم / عن / بعد / فالمغرررب 3awnatni / Technologies Corona / مزيان / كيخدمو / bzaaaf	in Morocco / distance learning / Greeting to helped me a lot / Technologies Corona / They work well
Transformation	التحية / للتعليم / عن / بعد / فالمغرررب 3awnatni / technologies التحية / للتعليم / عن / بعد / فالمغرررب عوناتني / تقنيات كورونا / مزيان / كيخدمو / بزاف	in morocco / distance learning / greeting to helped me a lot / technologies corona / they work well
Translation and transliteration		in morocco / distance learning / greeting to helped me a lot / technologies corona / they work well
Removing stop words	التحية / للتعليم / فالمغرررب عوناتني / تقنيات كورونا / مزيان / كيخدمو / بزاف	in morocco / distance learning / greeting to helped me a lot / technologies corona / they work well
Stemming	تحية / تعلم / مغرب عوناتني / تقنية كورونا / مزيان / كيخدمو / بزاف	morocco / learning / greeting to helped me a lot / technologie corona / they work well
Unigram	تحية عوناتني مزيان / كيخدمو / بزاف	greeting to helped me a lot they work well

Feature Extraction

After the different phases of pre-processing and cleaning, the sentences will mapped into a sentiment vector, which links the sentiment values of the words to calculate the sentiment of the studied sentences. We prepare the most relevant features for the task of classification by removing irrelevant and noisy data.

In this work, we have used unigrams as feature extractors. Several types of research have shown excellent performances of the unigram model in terms of sentiment detection, which motivates the choice of this model in this study [14]. The creation of the vector, the weight of the word given according to the text containing this word. There are several weighting methods such as Binary weighting; Term Frequency (TF) weighting; Term Frequency-Inverse Document Frequency (TFIDF); and Inverse Document Frequency (IDF) weighting. Binary weighting (regardless the fact whether it exists or not) was used in this paper by determining the weight of every word using a binary model to ensure that a word gets '1' if it is

Table 6 The confusion matrix

	Predicted class positive	Predicted class negative
Actual class positive	True Positive (TP)	False Negative (FN)
Actual class negative	False Positive (FP)	True Negative (TN)

present in the phrase. Otherwise, the word gets '0' [14]. Efficiency will be measured using the Accuracy (1) of each classifier, which is defined as:

$$\text{Accuracy} = \frac{\text{TN+TP}}{\text{TN+TP+FP+FN}} \tag{1}$$

Accuracy is defined as all true predicted cases against all predicted cases. If we receive 100% accuracy, it denotes that the predicted cases are precisely the same as the actual cases [15]. Table 6 displays the confusion matrix.

4 Experiments and Results

To conduct a thorough examination of the opinions held by Moroccans towards distance learning and its circumstances, this part introduces different machine learning algorithms, to ultimately identify the most efficient algorithm through comprehensive analysis.

This part begins by using the results that obtained during feature selection based on the datasets studied in this research. For classification, support vector machines (SVM) and Naïve Bayes (NB) are used. The SVM algorithm uses a Support Vector, Linear Support Vector and Nu-Support Vector, and for the NB algorithm, Multinomial Naïve Bayes and Bernoulli Naive Bayes are used.

A support vector machine (SVM) constructs a hyperplane or a set of hyperplanes in a high-dimensional or infinite-dimensional space, offering applications in tasks like classification or regression. The fundamental aim is to establish a robust separation, and this is achieved by identifying the hyperplane with the greatest distance to the nearest training data points for each class, termed the functional margin. The intuition behind this approach is that a larger margin generally corresponds to a lower generalization error for the classifier [16] in the following, we explain those algorithms.

4.1 Support Vector Machine Algorithms

Hereafter, we define three classes of algorithms:

– Support Vector

SVC stands for Support Vector Classification and is implemented based on LIBSVM (LIBSVM is a popular open-source machine learning library). The probability model of SVC is created using cross-validation so that the results can differ from those obtained by prediction. Moreover, it will produce irrelevant results in very small datasets. The SVC multiclass support implemented according to a one-vs-one scheme; SVC used for classifying linear and nonlinear data [16].

– Linear Support Vector

Support vector classifiers attempt to construct a hyperplane that maximizes the distance between various classes. Linear SVC, or Linear Support Vector Classification, is an implementation similar to SVC but using a linear kernel, that is, it tries to divide classes with a line. These classifiers are quite effective when there is a high dimensional space. The linear classifier implemented in the Scikit-learn library is used [17].

– Nu-Support Vector

It is similar to the basic SVC classifier but uses a parameter to control the number of support vectors. As in the previous case, the classifier implemented in the Scikit-learn library is used [17].

4.2 Naive Bayes Algorithms

– Multinomial Naive Bayes

This multinomial naive Bayes classifier expanded on the use of the naive Bayes (NB) algorithm. It is appropriate for discrete feature classification multinomial distributed data using multinomial NB. The multinomial distribution requires integer feature counts; however, fractional counts such as the TF-IDF work well in practice [18].

– Bernoulli Naive Bayes

This classifier is suitable for discrete data such as Multinomial NB, but Bernoulli was designed for binary or Boolean features. It uses naive Bayes (NB) for a multivariate Bernoulli distribution of data. Thus, every class needs samples, which have to be represented in binary values [18]. Also, Bernoulli NB can convert inputs of any other kind of data to binary. Table 6 shows the accuracy obtained for each topic using these algorithms.

– Training of Machine Learning Models

The successful implementation and training of our machine learning models were predominantly facilitated by leveraging the robust capabilities of the Scikit-Learn library. The critical tasks of vectorization and training were seamlessly imported from the library's sub-modules. Within our implementation, we employed various

models, primarily based on Support Vector Machines (SVM) and Naïve Bayes, all orchestrated within a Python script.

– Testing of Machine Learning Models

In this scenario, we collected a Test/Unlabeled Dataset from Twitter related to the same topics and we processed them in our Framework. Figure 4 shows results obtained by analysing people's opinions from the same period. Tables 7 and 8 reveals the accuracies achieved through various algorithms (with and without the application of Translation/Transliteration techniques). The results of these assessments lead us to a confident inference: the SVM algorithm outperforms others, demonstrating accuracies of 89.39, 90.46, and 91.37 for the 3 respective topics (when integrating translation/transliteration techniques).

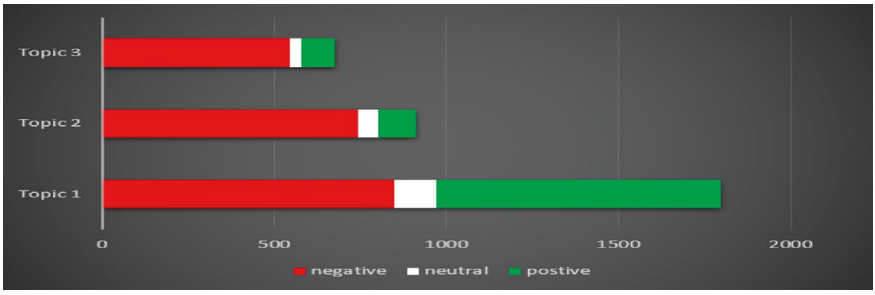


Fig. 4 Opinions obtained based on the trained model

Table 7 Results obtained with the three topics without applying translation/transliteration

Algorithms	Accuracy		
	1st topic	2nd topic	3rd topic
Support vector	79.10	89.00	88.14
Linear support vector	81.39	89.05	90.22
Nu-support vector	86.05	88.12	88.97
Multinomial Naive Bayes	83.42	84.99	82.45
Bernoulli Naive Bayes	77.19	83.19	85.07

Table 8 Results obtained with the three topics with applying translation/transliteration

Algorithms	Accuracy		
	1st topic	2nd topic	3rd topic
Support vector	88.60	89.04	88.95
Linear support vector	89.39	90.46	91.37
Nu-support vector	88.08	88.78	87.13
Multinomial Naive Bayes	83.87	85.91	83.68
Bernoulli Naive Bayes	79.77	84.93	85.95

5 Conclusions

In this article, we presented a hybrid approach to translation and transliteration from foreign languages and from Arabizi into Arabic. The latter based on a combination of the rules of statistical models to extract the best candidate to replace the Arabizi. We also applied several machine-learning algorithms to detect the best fit (SVM and NB); the results are relatively good, with an accuracy between 80 and 91%. The SVM algorithm gave better accuracy using Linear Support Vector Classification. In addition, the article evaluates the accuracies with and without applying translation/transliteration techniques. The findings demonstrate that integrating translation and transliteration techniques yields superior results compared to the baseline and improves the overall accuracy of the system. Following the analysis of the opinions of Moroccans, we note a great convergence between positive and negative opinions on the first subject because several Moroccan families find themselves confused between sending their children to school and exposing them to distance learning. On the other hand, many parents refused to pay school fees during the quarantine. Moreover, private schools said distance learning was necessary. This justifies the increase in negative opinions on the last two subjects. Therefore, the obtained results accurately reflect the reality of the situation.

In the future, we are considering implementing a Moroccan stemmer that could improve the word reduction process. Additionally, using bigram and n-gram features could increase results.

References

1. Liu, B.: Sentiment Analysis and Opinion Mining. Synthesis Lectures on Human Language Technologies, Vol. 5, no 1, pp. 1–167 (2012)
2. Alsaleem, S., et al.: Automated Arabic text categorization using SVM and NB. Int. Arab. J. Technol. **2**(2), 124–128 (2011)
3. Doniagamal, M.A., El-Horbaty, E.M., Salem, A.M.: Opinion mining for Arabic dialects on Twitter. Egypt. Comput. Sci. J. **42**(4) (2018)
4. Addawood, A., Alsuwailem, A., Alohali, A., Alajaji, D., Alturki, M., Alsuhaibani, J., Aljabli, F.: Tracking and understanding public reaction during COVID-19: Saudi Arabia as a use case. In: Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020 (2020)
5. Al-Smadi, M., Qawasmeh, O., Al-Ayyoub, M., Jararweh, Y., Gupta, B.: Deep recurrent neural network vs. support vector machine for aspect-based sentiment analysis of Arabic hotels' reviews. J. Comput. Sci. **27**, 386–393 (2018)
6. Larkey, L.S., Ballesteros, L., Connell, M.E.: Light stemming for Arabic information retrieval. In: Text, Speech and Language Technology Arabic Computational Morphology, pp. 221–243 (2007)
7. Khoja, S., Garside, R.: Stemming Arabic text. Lancaster, UK, Computing Department, Lancaster University (1999)
8. Bakshi, R.K., Kaur, N., Kaur, R., et al.: Opinion mining and sentiment analysis. In: 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), pp. 452–455. IEEE (2016)

9. Darwish, K.: Arabizi detection and conversion to Arabic. In: Proceedings of the EMNLP 2014 Workshop on Arabic Natural Language Processing (ANLP) (2014)
10. Roesslein, J.: Getting started—Tweepy 3.5.0 documentation (2009). [Online]. Available: http://docs.tweepy.org/en/v3.5.0/getting_started.html
11. Kunal, S., Saha, A., Varma, A., Tiwari, V.: Textual dissection of live twitter reviews using Naive Bayes. Procedia Comput. Sci. **132**, 307–313 (2018)
12. Loper, E., Bird, S.: NLTK: The Natural Language Toolkit. arXiv preprint cs/0205028 (2002)
13. Larkey, L.S., Connell, M.E.: Arabic information retrieval at UMass in TREC-10. In: TREC (2001)
14. Sharupa, N.A., Rahman, M., Alvi, N., Raihan, M., Islam, A., Raihan, T.: Emotion detection of twitter post using multinomial Naive Bayes. In: 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (2020)
15. Joshi, N.S., Itkat, S.A.: A survey on feature-level sentiment analysis. Int. J. Comput. Sci. Inform. Technol. **5**(4), 5422–5425 (2014)
16. Desai, M., Mehta, M.A.: Techniques for sentiment analysis of Twitter data: a comprehensive survey. In: 2016 International Conference on Computing, Communication and Automation (ICCCA) (2016)
17. Scikit-learn: Machine learning in Python. J. Mach. Learning Res. **12**, 2825–2830 (2011)
18. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval, pp. 234–265. Cambridge University Press, Cambridge (2008)

Artificial Intelligence in Education and eHealth

Leveraging Artificial Intelligence (AI)-Enhanced STEM Cognition-Multi-Directionality of Influence

Anass Bayaga

Abstract This research explores the integration of artificial intelligence (AI) in Science, Technology, Engineering, and Mathematics (STEM) cognition, focusing on predictive modeling and performance analysis. The study formulated hypothesis to address gaps identified in the existing literature on AI in educational contexts. Using Bootstrap MGA, Parametric Test, Welch Satterthwaite analyses, and Importance-Performance Map Analysis (IPMA), the research assessed gender differences and predictive importance of latent variables in the structural model. With a sample size of 71 students, the study employed rigorous testing for convergent and discriminant validities in the questionnaire design. Results indicated generally non-significant gender disparities, except for a potential gender-specific nuance in the MMS→ACP pathway. The IPMA highlighted Analogical Comparison Principle (ACP) as a robust predictor (total effect = 1), while Mathematical Cognition (MAS) showed low importance (total effect = 0.05). Mathematical and Computational Algorithms (MCA) emerged as a substantial predictor across gender groups (total effects ranging from 0.5 to 0.59), and Mathematical Modelling and Simulation (MMS) exhibited varying effects.

Keywords Artificial intelligence · STEM education · Predictive modeling · Gender differences

1 Background and Literature Review on: Leveraging AI-Enhanced STEM Education

The current research explores the integration of Adaptive E-learning Technologies in the form of artificial intelligence (AI) in Science, Technology, Engineering, and Mathematics (STEM) cognition, focusing on predictive modeling and performance analysis. The study formulated hypothesis to address gaps identified in the existing

A. Bayaga (✉)
University of the Western Cape, Gqeberha, South Africa
e-mail: abayaga@uwc.ac.za

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_20

literature on AI in educational contexts. In the field of leveraging AI and STEM education, number of interactive relationships have been argued for as a conceptual model [1, 3, 5–7]. Such conceptual model as recounted by literature [2, 4, 10] illustrates the interconnectedness of various components in AI-enhanced STEM education with various elements such as; Mathematical Cognition: Ease of Learning and Acquisition of Skills (MAS), Mathematical Modelling and Simulation Using Adaptive E-learning Technologies (MMS), Mathematical and Computational Algorithms (MCA), Abstract and Concrete Representations of Mathematical Principles (MP), Error and Misconceptions Reflection Principle (EMR), and Analogical Comparison Principle (ACP).

Collectively, evaluation of literature suggests both direct and indirect relations such as mediated and moderated (possible demographics such as gender) relations. The relationships (see Fig. 1) for instance, indicate the directionality of influence or interaction between the components. For example, in the proposed model, it was inferred from literature that MMS is posited to directly influence both MAS and MCA, suggesting that Mathematical Modelling and Simulation Using Adaptive E-learning Technologies is foundational to Mathematical and Computational Algorithms as well as the understanding of Mathematical and Computational Algorithms. Furthermore, the model proposes a bidirectional influence between MCA and MP, indicating a dynamic relationship between algorithmic processes and the representations of mathematical principles [7, 8].

Drawing from the existing research debates, the present study posits a set of comprehensive (multi-directionality of influence) hypothesis within the framework of Mathematical Cognition: Ease of Learning and Acquisition of Skills (MAS), Mathematical Modelling and Simulation Using Adaptive E-learning Technologies (MMS), Mathematical and Computational Algorithms (MCA), Abstract and Concrete Representations of Mathematical Principles (MP), Error and Misconceptions Reflection Principle (EMR), and Analogical Comparison Principle (ACP). The central proposition is that these principles exert varying influences on learners

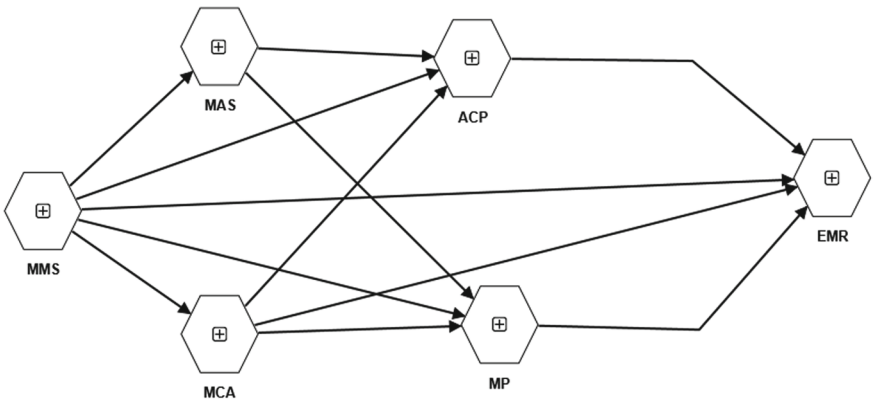


Fig. 1 Conceptual model for leveraging AI-enhanced STEM cognition

of distinct genders, contributing to diverse outcomes amid the implementation of Leveraging AI-Enhanced STEM Education [2, 9–11].

2 Research Methodology

2.1 Questionnaire Design

This study aimed to assess AI-enhanced STEM cognition through questionnaire survey from 71 students. The questionnaire, comprising two sections, explored participants' socio-demographic background based on literature. The second part delved into key constructs influencing error and misconceptions reflection principle (EMR) there by AI-enhanced STEM education. Responses were collected using a four-point Likert scale for increased discrimination and reliability.

2.2 Data Collection Process

The questionnaire for this study underwent rigorous testing for convergent and discriminant validity checks by academic after being administered to the 71 students utilizing probability random sampling. Data screening eliminated any invalid responses, and the analysis confirmed the sufficiency of the sample size, exceeding the minimum requirement of 50 [12, 13].

2.3 Ethical Consideration

This section outlines the ethical commitments. The work represents the original work of the author and has not been published elsewhere previously. Furthermore, it is not currently under consideration for publication elsewhere. The paper accurately reflects the author's research and analysis. To ensure the privacy and confidentiality of respondents, details of interviewees were anonymized. Ethical procedures included obtaining Institutional Review Board approval from XXX University, South Africa, with reference number H21-EDU-PGE-026. The research also received ethical approval from XXX University's Faculty of Education's research ethics committee, aligning with established norms and practices to safeguard the well-being of participants. Additionally, prior permission was obtained from the principals of Kwazulu-Natal colleges before commencing data collection.

2.4 Data Analysis

This study employed structural equation modelling (SEM) to simultaneously analyze hypothesized relationships in a model and potential direct and indirect relationship between multiple endogenous and exogenous variables [12]. Given the exploration of the modified conceptual model (see Fig. 1) drawn from literature, partial least squares structural equation modelling (PLS-SEM) was deemed more appropriate particularly multi group analysis (MGA). PLS-SEM through MGA-PLS has the advantage of requiring a lower sample size and accommodating non-normally distributed data compared to covariance-based SEM (CB-SEM) [12]. SmartPLS™ version 4 software was utilized for PLS-SEM to assess the measurement model and test path relationships between model constructs based on collected data [12]. To examine differences in hypothetical relationships between groups, a multi-group analysis (MGA) in PLS-SEM was conducted. The overall sample was divided into groups based on a categorical variable of interest. Measurement invariance (MICOM) across groups was then assessed in three steps: MICOM STEM2, MICOM STEP 3a(mean) and MICOM STEP 3b(variance) which account for configural invariance, compositional invariance, equality of composite mean values and variances [12]. After establishing measurement invariance, a comparison of path coefficients among groups using the Henseler PLS-MGA procedure was carried out to determine significant differences [12, 13].

3 Results

3.1 Assessment of Measurement Model

The MGA-PLS analysis yielded robust results in assessing the measurement model and validating the structural model. Reliability and convergent validity demonstrated satisfactory internal consistency for all constructs in the complete sample and gender subgroups. Cronbach's alpha (α) exceeded the recommended threshold of 0.7, Composite Reliability (CR) surpassed 0.7, and Average Variance Extracted (AVE) exceeded 0.5, indicating the reliability and convergent validity of the constructs. Discriminant validity, evaluated through the Heterotrait-Monotrait (HTMT) ratio, exhibited values below the recommended threshold of 0.85, confirming discriminant validity among the constructs [12].

In the Fornell-Larcker structural equation modeling analysis, the correlation matrix is presented for the complete sample as well as disaggregated by gender [13]. For the complete sample, the diagonal values represent the square root of the average variance extracted (AVE) for each latent variable, indicating the amount of variance explained by the respective constructs. The off-diagonal values signify the correlations between the latent variables. Notably, ACP (Analogical Comparison Principle)

shows a high average variance extracted (AVE) of 0.83, suggesting strong convergent validity. In the female subgroup, similar patterns are observed with high AVE for ACP (0.84) and EMR (Error and Misconceptions Reflection Principle) (0.81), reinforcing their convergent validity. The male subgroup, regarding ACP (0.84) and EMR (0.79) demonstrated robust AVE with following R-square values; ACP explains 59% of the variance in the complete sample, a slightly higher proportion for males (74%) compared to females (62%).

3.2 Assessment of Structural Model and PLS-MGA

Assessment of Variance Inflation Factor (VIF), the complete sample, and male as well as female subgroups VIF values are within an acceptable range, indicating no significant multicollinearity issues among the latent variables [12].

The direct paths in Table 1 reveal the strength and significance of relationships between key constructs in the Leveraging AI-Enhanced STEM Education model for the complete sample and gender subgroups. In the complete sample, the positive relationship between Analogical Comparison Principle (ACP) and Error and Misconceptions Reflection (EMR) is robust ($\beta = 1$, $T = 10.6$, $p < 0.001$), emphasizing the importance of connecting or noticing contrasting features for better understanding and reflection on errors. However, the direct path from Mathematical Cognition: Ease of Learning and Acquisition of Skills (MAS) to ACP is not significant ($\beta = 0.07$, $T = 0.86$, $p = 0.39$), suggesting that the fluency in learning mathematics may not directly influence analogical comparison. Notably, the direct path from MAS to Mathematical and Computational Algorithms (MCA) is significant ($\beta = 0.67$, $T = 9.47$, $p < 0.001$), highlighting the role of mathematics fluency in understanding computational algorithms. The gender-specific analyses provide nuanced insights into the relationships, emphasizing the need for tailored approaches in STEM education.

4 Meditated Effects: Total Indirect Effects and Specific Indirect Effects

The study investigated mediating effects and total indirect effects within the Leveraging AI-Enhanced STEM cognition model, examining gender-specific variations. In the complete sample, mediating effects reveal that Mathematical and Computational Algorithms (MCA) significantly mediate the relationship between Mathematical Modelling and Simulation (MMS) and Error and Misconceptions Reflection (EMR) ($\beta = 0.71$, $p < 0.001$).

Table 2 was used to assess the Measurement Invariance of Composite Models (MICOM) before the MGA PLS, thus examining the robustness and consistency of the model across groups. The correlation coefficients for ACP, EMR, MAS, MCA,

Table 1 Direct relationship

	Complete				Female				Male			
	Beta	SD	T values	P values	Beta	SD	T values	P values	Beta	SD	T values	P values
Direct paths												
ACP→EMR	1	0.09	10.6	0	1.04	0.17	6.12	0	0.91	0.22	4.05	0
MAS→ACP	0.07	0.08	0.86	0.39	0.05	0.13	0.4	0.69	0.04	0.1	0.44	0.66
MAS→MP	−0.18	0.11	1.62	0.11	−0.25	0.19	1.33	0.18	−0.19	0.17	1.08	0.28
MCA→ACP	0.67	0.07	9.47	0	0.75	0.08	9.33	0	0.71	0.12	5.88	0
MCA→EMR	−0.21	0.13	1.57	0.12	−0.21	0.25	0.86	0.39	−0.13	0.21	0.63	0.53
MCA→MP	0.41	0.15	2.65	0.01	0.44	0.28	1.56	0.12	0.43	0.19	2.29	0.02
MMS→ACP	0.28	0.08	3.37	0	0.13	0.12	1.07	0.28	0.44	0.11	3.93	0
MMS→EMR	−0.24	0.09	2.71	0.01	−0.33	0.1	3.17	0	−0.17	0.2	0.84	0.4
MMS→MAS	−0.12	0.11	1.08	0.28	−0.22	0.15	1.49	0.14	0.05	0.16	0.31	0.76
MMS→MCA	0.14	0.13	1.02	0.31	0.2	0.21	0.94	0.35	0.07	0.17	0.39	0.7
MMS→MP	−0.11	0.11	0.96	0.34	−0.22	0.23	0.96	0.34	0.05	0.17	0.29	0.77
MP→EMR	0.12	0.09	1.35	0.18	0.06	0.13	0.49	0.63	0.13	0.17	0.8	0.43

MMS, and MP analysed and demonstrate strong associations, with ACP-EMR ($r = 1$, $p = 0.01$) and EMR-MAS ($r = 0.99$, $p = 0.02$) showing significant and stable relationships with permutation mean differences, confidence intervals (2.50–7.50%), and percentile values affirming the stability of these correlations, indicating consistent results across permutations. The results underscore the reliability and generalizability of the MGA PLS SEM model across diverse groups. The results established the MICOM, ensuring its applicability across different contexts.

Table 3, as reflected via Bootstrap MGA, Parametric Test, Welch Satterthwaite, examined significant patterns, thus observed for MAS→ACP, MAS→MP, MCA→ACP, MCA→EMR, MCA→MP, MMS→ACP, MMS→EMR, MMS→MAS, MMS→MCA, MMS→MP, and MP→EMR, with non-significant differences and consistent p-values across the three analytical approaches. For example, in the path from ACP to EMR ($\beta = -0.13$, $p = 0.63$), the difference is not statistically significant. Similar non-significant differences are observed for other paths, such as MAS to ACP ($\beta = -0.01$, $p = 0.97$), MAS to MP ($\beta = 0.06$, $p = 0.80$), MCA to ACP ($\beta = -0.04$, $p = 0.84$), MCA to EMR ($\beta = 0.08$, $p = 0.80$), MCA to MP ($\beta = -0.01$, $p = 0.98$), MMS to ACP ($\beta = 0.31$, $p = 0.06$), MMS to EMR ($\beta = 0.16$, $p = 0.48$), MMS to MAS ($\beta = 0.27$, $p = 0.22$), MMS to MCA ($\beta = -0.13$, $p = 0.63$), MMS to MP ($\beta = 0.27$, $p = 0.34$), and MP to EMR ($\beta = 0.07$, $p = 0.74$).

5 Importance Performance Map Analysis for EMR

The Importance Performance Map Analysis (IPMA) in Table 4 assesses the total effects and performance of each latent variable (ACP, MAS, MCA, MMS, MP) in the model across different groups (Complete, Female, Male). For ACP, it exhibits a substantial total effect in all groups, with the highest importance in the male group (importance = 0.91, performance = 49.6). Similarly, MAS demonstrates a modest total effect in all groups, with slightly higher importance and performance in the female group (importance = 0.04, performance = 23.81).

As reflected in both Table 4 and Fig. 2, MCA shows a significant total effect, particularly in the male group (importance = 0.57, performance = 23.2). MMS displays varying effects, with a negative total effect in the female group (importance = −0.11, performance = 33.33) and positive effects in the other groups. Finally, MP has a moderate total effect in all groups, with the male group showing the highest importance (importance = 0.13, performance = 23.5).

Table 2 MICOM for MGA PLS SEM

MICOM STEP2					MICOM STEP3a(mean)					MICOMSTEP3b(variance)				
Constructs	Original correlation	Correlation permutation mean	5.00%	Permutation p value	Original difference	Permutation mean difference	2.50%	97.50%	Permutation p value	Original difference	Permutation mean difference	2.50%	97.50%	Permutation p value
ACP	1	1	1	0.25	0.19	0.01	−0.46	0.48	0.43	0.09	0	-0.6	0.57	0.73
EMR	1	1	0.99	0.52	0.37	0.02	−0.46	0.46	0.12	0.53	0.01	-0.65	0.71	0.12
MAS	1	1	1	0.04	−0.08	0.01	−0.47	0.5	0.8	-0.2	0.01	-0.82	0.91	0.72
MCA	0.99	0.99	0.98	0.17	−0.34	0	−0.46	0.47	0.15	-0.4	-0.01	-0.9	0.84	0.45
MMS	1	1	1	0	−0.03	0	−0.51	0.44	0.84	-0.3	0.01	-0.61	0.61	0.42
MP	0.99	0.96	0.83	0.48	−0.28	0	−0.47	0.46	0.27	-0.2	-0.01	-1.21	1.24	0.82

Table 3 Bootstrap MGA, parametric test, Welch Satterthwaite

Bootstrap MGA				Parametric test			Welch Satterthwaite		
Paths	Difference (male—female)	1-tailed (male vs female) p value	2-tailed (male vs female) p value	Difference (male—female)	t value (|male vs female|)	p value (male vs female)	Difference (male–female)	t value (|male vs female|)	p value (male vs female)
ACP→EMR	−0.13	0.69	0.62	−0.13	0.48	0.63	−0.13	0.48	0.63
MAS→ACP	−0.01	0.52	0.97	−0.01	0.04	0.97	−0.01	0.04	0.97
MAS→MP	0.06	0.38	0.77	0.06	0.25	0.8	0.06	0.25	0.8
MCA→ACP	−0.04	0.58	0.84	−0.04	0.27	0.78	−0.04	0.27	0.79
MCA→EMR	0.08	0.4	0.8	0.08	0.25	0.8	0.08	0.25	0.8
MCA→MP	−0.01	0.54	0.91	−0.01	0.03	0.98	−0.01	0.03	0.98
MMS→ACP	0.31	0.03	0.06	0.31	1.94	0.06	0.31	1.94	0.06
MMS→EMR	0.16	0.25	0.49	0.16	0.72	0.47	0.16	0.71	0.48
MMS→MAS	0.27	0.11	0.22	0.27	1.26	0.21	0.27	1.26	0.22
MMS→MCA	−0.13	0.7	0.61	−0.13	0.48	0.63	−0.13	0.48	0.63
MMS→MP	0.27	0.16	0.33	0.27	0.96	0.34	0.27	0.97	0.34
MP→EMR	0.07	0.37	0.73	0.07	0.33	0.74	0.07	0.33	0.74

Table 4 Importance performance map analysis for EMR-IPMA

s			Female		Male	
	Total effects (importance)	Performance	Total effect (importance)	Performance	Total effect (importance)	Performance
ACP	1	46.24	1.04	43.7	0.91	49.6
MAS	0.05	23.47	0.04	23.81	0.02	21.9
MCA	0.5	26.89	0.59	30.55	0.57	23.2
MMS	0.09	31.34	−0.11	33.33	0.27	32.3
MP	0.12	26.91	0.06	30.1	0.13	23.5

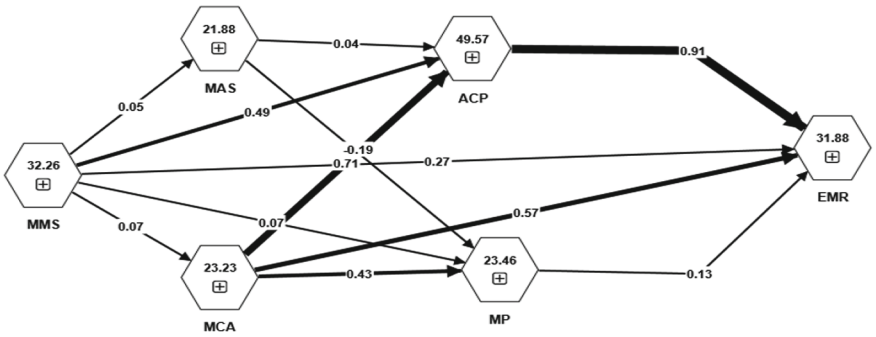


Fig. 2 Importance performance map analysis for leveraging AI-enhanced STEM cognition

6 Discussions

The Leveraging AI-Enhanced STEM Education model underwent a comprehensive evaluation through MGA-PLS analysis, affirming its robustness and reliability. The measurement model assessment revealed high internal consistency across constructs, surpassing recommended thresholds for Cronbach's alpha ($\alpha > 0.7$), Composite Reliability (CR > 0.7), and Average Variance Extracted (AVE > 0.5). Discriminant validity was confirmed through Heterotrait-Monotrait ratios and the Fornell-Larcker criterion, attesting to the distinctiveness of the constructs. The mediation analysis unveiled significant indirect effects, shedding light on the intricate relationships among latent variables. Collinearity assessment, employing Variance Inflation Factor (VIF), and direct path analysis underscored the absence of multicollinearity and elucidated the strength and significance of relationships between key constructs. Gender-specific analyses yielded nuanced insights, emphasizing the need for tailored interventions in male and female STEM education settings. Measurement invariance analysis (MICOM) demonstrated stability across permutations, reinforcing the reliability of the MGA PLS SEM framework. Bootstrap MGA and Welch Satterthwaite analyses indicated no statistically significant differences between male and female groups in structural paths (ACP→EMR, MAS→ACP,

MAS→MP, MCA→ACP, MCA→EMR, MCA→MP, MMS→ACP, MMS→EMR, MMS→MAS, MMS→MCA, MMS→MP, MP→EMR), confirming the consistency of relationships across gender groups. The Importance Performance Map Analysis (IPMA) highlighted the pivotal role of Analogical Comparison Principle (ACP) across all groups (Complete, Female, Male), providing valuable insights for theoretical refinement and practical application in AI-enhanced STEM education research and interventions.

Thus, the assessment of the measurement model reveals the robustness of latent constructs (ACP, EMR, MCA, MP) within and across genders. Reliability coefficients (α) exhibit high internal consistency, surpassing the recommended threshold of 0.80. Composite reliability (CR) exceeds 0.88, indicating satisfactory reliability, and the average variance extracted (AVE) demonstrates adequate convergent validity (>0.64). These findings align with established criteria for construct reliability and validity [12]. Discriminant validity, assessed through Heterotrait-Monotrait (HTMT) ratios, showcases constructs' higher correlations with themselves than with others, affirming robust discriminant validity.

7 Theoretical Implication

The results of the direct and mediated effects within the hypothesized model are presented in Table 3, offering insights into the complex interplay between Mathematical Cognition principles and gender-specific learning outcomes. In the direct paths, the beta coefficients reveal significant associations. Notably, ACP→EMR exhibits a robust and positive relationship across all groups (Complete, Female, Male), with beta values of 1, 1.04, and 0.91, respectively, indicating that an increase in the Analogical Comparison Principle is associated with a corresponding increase in Error and Misconceptions Reflection. Similar positive associations are observed for MCA→ACP, MCA→MP, MMS→ACP, and MP→EMR, suggesting the influential role of these factors in the context of leveraging AI-Enhanced STEM Education. In contrast, MAS→MP shows a negative relationship, implying that an increase in Mathematical Cognition may be associated with a decrease in Misconceptions Reflection, but the significance varies across genders. In summary, ACP appears to be crucial across all groups, while MAS, MCA, MMS, and MP contribute to varying degrees. The IPMA provides insights into the relative importance and performance of each latent variable in the model, aiding in the identification of key drivers and areas for improvement.

Moving to the mediating effects, the total indirect effects shed light on the overall impact of specific pathways. MCA→EMR stands out with a substantial beta of 0.71 in the Complete group, indicating that Mathematical and Computational Algorithms have a considerable indirect influence on Error and Misconceptions Reflection.

8 Conclusion

Notably, the analyses did reveal a potentially interesting result for MMS→ACP, where a difference of 0.31 was observed, with a p-value of 0.06 in the bootstrap MGA, suggesting a marginally significant difference. However, this result was not replicated in the parametric tests or Welch Satterthwaite comparisons. These findings provide insights into gender differences across latent variables, highlighting both non-significant trends and potential nuances in the relationship between Mathematical Cognition and the use of AI-Enhanced STEM Education. In conclusion, the results suggest that there are no statistically significant differences between male and female groups in the structural paths, as all p values for the differences in these paths exceed the 0.05 threshold. This indicates that the relationships among latent variables are consistent across gender groups in the model.

References

1. Baker, T., Smith, L., Anissa, N.: Education rebooted? Exploring the Future of Artificial Intelligence in Schools and Colleges (2019). Retrieved from https://www.nesta.org.uk/report/education-rebooted/
2. Chichekian, T., Benteux, B.: The potential of learning with (and not from) artificial intelligence in education. Front. Artif. Intell. 5–20 (2022)
3. Fan, O., Jiao, P.: Artificial intelligence in education: the three paradigms. Comput. Educ. Artif. Intell. **2**(1), 10–20 (2021)
4. Harrer, A., Joolingen, W.R., Hoppe, H.U.: Modelling for learning tasks—approaches, tools and implications. In: IEEE International Conference on Advanced Learning Technologies. Proceedings, 1086–1087 (2004)
5. Hwang, G., Tu, Y., Tang, K.: AI in online-learning research: visualizing and interpreting the journal publications from 1997 to 2019. Int. Rev. Res. Open Distrib. Learning **23**(1), 25–46 (2022)
6. Hwang, G., Xie, H., Wah, B.W., Gašević, D.: Vision, challenges, roles and research issues of artificial intelligence in education. Comput. Educ. Artif. Intell. **1**(1), 100–120 (2020)
7. Liao, J., Yang, J., Zhang, W.: The student-centered STEM learning model based on artificial intelligence project: a case study on intelligent car. Int. J. Emerg. Technol. Learn. **16**(1), 53–68 (2021)
8. Krstić, L., Aleksić, V., Krstić, M.: Artificial intelligence in education: a review. Proceedings TIE 2022 **11**(2), 243–248 (2022)
9. Rose, C.P., McLaughlin, E.A., Liu, R., Koedinger, K.R.: Explanatory learner models: why machine learning (alone) is not the answer. Br. J. Edu. Technol. **50**(6), 2943–2958 (2019)
10. Xie-Li, D., Arias-Méndez, E.: Artificial intelligence in stem education: interactive hands-on environment using open source electronic platforms. Tecnología en Marcha **36**(Special Issue), 45–52 (2023). https://doi.org/10.18845/tm.v36i6.6759
11. Zawacki-Richter, O., Marín, V.I., Bond, M., Gouverneur, F.: Systematic review of research on artificial intelligence applications in higher education—where are the educators? Int. J. Educ. Technol. High. Educ. **16**, 1–27 (2019)
12. Hair, J.F., Hult, G.T.M., Ringle, C.M., Sarstedt, M.: A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM), 3rd edn. Sage, Thousand Oaks, CA (2022)
13. Fornell, C., Larcker, D.F.: Evaluating structural equation models with unobservable variables and measurement error. J. Mark. Res. **18**(1), 39–50 (1981). https://doi.org/10.2307/3151312

Artificial Intelligence and Assessment Generators in Education: A Comprehensive Review

Youness Boutyour, Abdellah Idrissi, and Lorna Uden

Abstract Artificial Intelligence (AI) has made substantial strides within educational assessment generators. This comprehensive review rigorously investigates the multifaceted impact of AI on assessment generators in education, encompassing the entire spectrum from examination creation to practical implementation. The study elucidates the underpinning theories and models that drive AI-assisted assessment generators, critically evaluating their inherent strengths, limitations, and ethical considerations. Through a meticulous examination of relevant research studies, this review affords an intricate portrayal of how AI seamlessly integrates into the realm of assessment generators. The outcomes underscore the transformative potential of AI in enhancing the quality and efficiency of assessments while also recognizing the imperative concerns regarding biases and technological constraints. Ultimately, this investigation culminates by charting a path forward, suggesting avenues for further research and practical applications. In summary, this study delves into the paradigm shift AI instigates within educational assessment practices through its pivotal role in assessment generators.

Keywords Artificial Intelligence (AI) · Assessment Generators · Education · Assessment Design · Knowledge Spaces · Learning Efficiency

Y. Boutyour (✉) · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team Faculty of Science, Mohammed V University in Rabat, Rabat, Morocco
e-mail: youness.boutyour@um5r.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

L. Uden
School of Computing, FCDT, Staffordshire University, Stoke-On-Trent, UK
e-mail: l.uden@staffs.ac.uk

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_21

1 Introduction

Artificial Intelligence (AI) stands at the forefront of transformative technologies, poised to redefine a myriad of sectors, notably education. Within this paradigm, AI-driven assessment generators emerge as pivotal tools, promising to reshape educational evaluation processes. This paper delves into the burgeoning intersection of AI and assessment generation, offering a comprehensive literature review that navigates through the complexities and potential of this integration.

We commence by charting the evolution and historical context of both AI and assessment generators, providing a foundation for understanding their convergence. The paper systematically explores various frameworks and models that underpin AI-supported assessment generators, scrutinizing their theoretical and practical underpinnings. A critical analysis of the benefits and limitations associated with AI in assessment generation forms a core component of this study, where we rigorously address prevailing concerns regarding biases and technological constraints.

Employing a meticulously defined search methodology, the review selects and scrutinizes relevant research, thereby contributing significantly to the academic discourse on AI's application in education. This investigation not only synthesizes existing studies on AI-driven assessment but also identifies critical literature gaps. It ventures further to propose prospective research trajectories, emphasizing the need for innovative methods and analyses.

Moreover, this paper foregrounds the transformative impact of technology on educational assessment methods, probing into its broader societal implications. Through a nuanced evaluation, we aim to illuminate the multifaceted role of AI in refining assessment tools, thereby influencing both the development and application of educational assessments. This research aspires to present a balanced and insightful perspective on how AI is not just augmenting but also redefining assessment methodologies in education.

2 Literature Review

2.1 Definition of AI and Assessment Generators

Artificial intelligence (AI) refers to the field of computer science that focuses on creating computers to perform tasks that typically require human intelligence like learning, problem-solving, and decision-making [1]. In the realm of assessment generators AI technology is utilized to develop algorithms that can generate assessments evaluate responses and provide feedback to learners [2].

On the other hand, assessment generators are tools used by educators to create types of assessments such, as quizzes, exams, and assignments. These tools streamline the process of assessment development while offering a range of question formats

and options. This does not save time. Also enhances efficiency in assessment creation [3].

The integration of AI with assessment generators has the potential to revolutionize educational assessment practices. It can facilitate adaptive assessments while enhancing their quality and efficiency. Additionally, it can alleviate the burden, on educators by reducing their workload [3, 4]. However, it is important to consider social implications when employing AI in assessment generators. These concerns include biases and privacy issues [5, 6].

2.2 History and Development of AI in Assessment Generators

The application of intelligence (AI), in assessment generators has had an impact despite its relatively short history. In the stages of AI assessment generators were limited to multiple-choice questions stored in item banks. However, as AI algorithms have advanced these generators have become more sophisticated and adaptable. Using natural language processing techniques (NLP) was designed to grade essays based on their content and organization. This marked a milestone in applying AI to automate evaluation tasks.

Since then, the use of AI in assessment generators has continued to evolve with the development of algorithms, like machine learning and deep learning. These advancements have enabled assessment generators to create assessments and offer personalized feedback tailored to individual learners. This has the potential to revolutionize how assessments are designed, administered, and evaluated within settings.

However, as AI-based evaluation generators become more prevalent concerns arise regarding biases and technological limitations that may need consideration.

AI algorithms for instance might have biases, towards student groups. Struggle to capture the full extent of students' skills and knowledge [7]. That's why it's crucial to examine the history and evolution of AI in assessment tools and consider the social implications they bring about.

2.3 Theoretical Frameworks and Models for AI-Assisted Assessment Generators

The development of AI-assisted assessment generators has been driven by a range of theoretical frameworks and models, which offer a structured approach to benefit from AI for enhancing assessment design, delivery, and evaluation. These frameworks and models provide valuable guidance for the development of AI-assisted assessment generators. By understanding these frameworks and models, educators and assessment designers can make informed decisions about how to use AI to support

assessment and learning. In this section, we address some of these frameworks with real-world case studies to clarify their practical applications.

Cognitive Diagnostic Assessment

The cognitive diagnostic assessment model (CDA) is one of the most extensively utilized theoretical frameworks for AI-assisted assessment generators [8–11]. CDA is founded on the principle that assessments should not only evaluate overall performance but also highlight individual strengths and weaknesses in learners' knowledge and abilities. Learners' responses to assessment questions may be analyzed using AI algorithms to find patterns of correct and incorrect replies, which can subsequently be used to diagnose particular learning challenges and offer tailored feedback. Figure 1 illustrates the conceptual framework for diagnostic assessments, showcasing the development of a CDA model based on the Assessment Triangle and its application in Cognitive Diagnostic Models (CDMs).

Case Study: Personalized Learning Path:

In a university setting, an AI-assisted assessment generator implemented CDA principles to create personalized learning paths for each student. The system identified knowledge gaps by analyzing students' responses to diagnostic questions and customized a learning plan. As a result, students received targeted resources and assessments tailored to their specific needs, significantly improving their learning outcomes.

Personalized Adaptive Learning Paradigm

The Personalized Adaptive Learning (PAL) paradigm is another theoretical framework for AI-assisted assessment generators [12, 13]. PAL is predicated on the notion

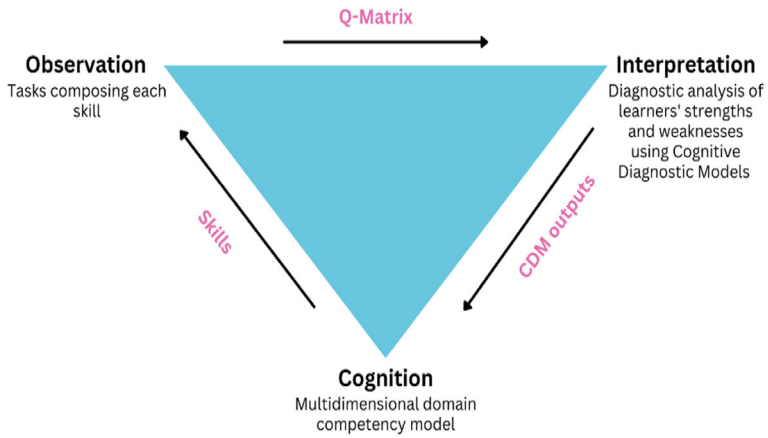


Fig. 1 Development of a CDA theoretical framework based on the assessment triangle for diagnostic assessments applying CDMs

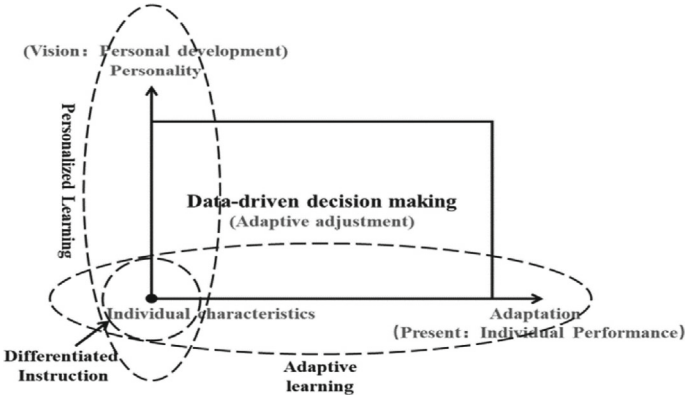


Fig. 2 Personalized adaptive learning framework [15]

that different learners have distinct learning needs and preferences and that assessments should be adapted to these needs and interests [14]. AI algorithms can be used to examine performance data from learners and develop adaptive assessments that provide individualized feedback and support. Figure 2 illustrates the Personalized Adaptive Learning framework.

Case Study: Adaptive Online Math Tutor:

An online math education platform implemented the PAL paradigm to create an adaptive learning experience. Students' interactions with the system were continuously analyzed, allowing it to adapt the difficulty of questions and provide real-time hints and explanations based on their performance. This adaptive approach resulted in increased engagement and motivation among students, leading to improved learning outcomes.

Knowledge Space Theory Model

The knowledge space theory (KST) model proposes that assessments should be designed to cover a range of knowledge domains and assess learners' understanding of these domains. AI algorithms can be used to generate assessments that cover multiple knowledge domains and adapt to learners' performance levels [16–19]. Figure 3. illustrates an example of key components of the knowledge domain relations.

Case Study: multidisciplinary Assessment for K-12 Education:

In a K-12 education context, AI-assisted assessment generators based on KST principles were employed to create multidisciplinary assessments. These assessments encompassed knowledge domains from multiple subjects and dynamically adjusted the level of difficulty based on students' responses. This approach encouraged holistic learning and cross-disciplinary thinking among students.

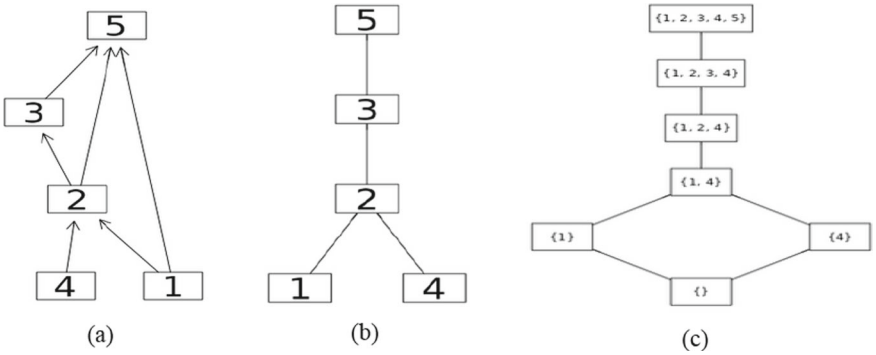


Fig. 3 **a** Relations between knowledge domains. **b** Surmise relation. **c** Hasse diagram of the corresponding knowledge space

Bayesian Network Model

The Bayesian network model is another option for AI-assisted assessment generators [20–22]. This concept is founded on the premise that assessments should be structured to measure not just what learners know, but also how they reason, and form conclusions [23]. AI algorithms may be used to develop assessments that test learners' reasoning and inferential skills and offer feedback on how to improve these skills. Figure 4. presents an example of the Bayesian Network 'Student Model' as described in the pioneering work of Gordon et al. [24].

Case Study: critical Thinking Assessment:

A higher education institution applied the Bayesian Network Model to create assessments focused on critical thinking skills. AI-assisted assessment generators analyzed students' responses to complex scenarios, evaluating not only their knowledge but

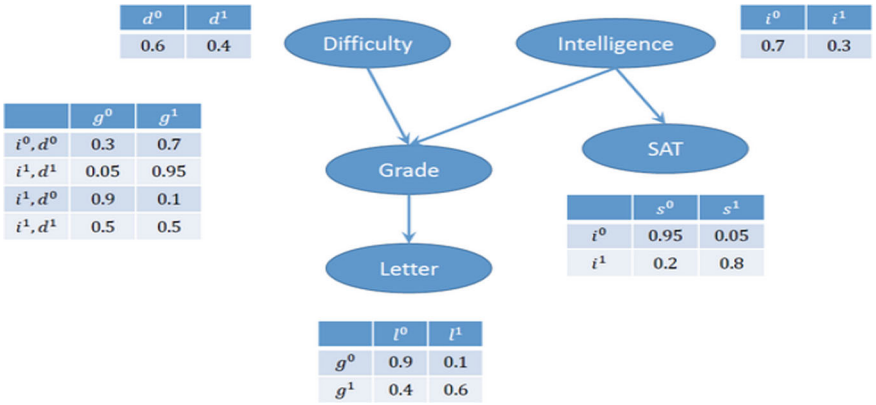


Fig. 4 Example of Bayesian network 'Student Model' [24]

also their ability to make reasoned judgments. Students received feedback on their critical thinking processes, enhancing their analytical skills and decision-making abilities.

2.4 Advantages and Limitations of AI-Based Assessment Generators

AI-based assessment generators provide some benefits over traditional assessment techniques, but they have several limits that must be considered.

Advantages:

One of the main advantages of AI-based assessment generators is their ability to provide personalized feedback to learners [12]. By analyzing learners' performance data, AI algorithms can provide targeted feedback on specific areas where learners need improvement, which can help to improve learning outcomes.

Another advantage of AI-based assessment generators is their ability to generate adaptive assessments that are tailored to learners' needs [25]. AI algorithms can analyze learners' performance data and generate assessments that provide an appropriate level of challenge and support, which can help to keep learners engaged and motivated.

AI-based assessment generators also offer the potential for greater efficiency and cost-effectiveness compared to traditional assessment methods [26, 27]. AI algorithms can analyze large amounts of assessment data quickly and accurately, which can reduce the time and resources required for assessment design and delivery.

Limitations:

One of the main limitations of AI-based assessment generators is their potential for bias and error [26]. AI algorithms may be programmed with biases that reflect the perspectives and values of their developers, which can lead to unfair or inaccurate assessments. In addition, AI algorithms may make errors in the analysis and interpretation of assessment data, which can lead to incorrect feedback and evaluations.

Another issue with AI-based assessment generators is that they may have worse validity and reliability than traditional evaluation techniques [25, 28]. AI algorithms may be unable to capture the entire spectrum of information, skills, and talents being examined, resulting in incomplete or erroneous judgments. Moreover, AI-based evaluations may be unable to account for contextual aspects such as emotional or social factors that may impact learners' performance.

Notwithstanding these limitations, AI-based assessment generators have the potential to significantly improve assessment design, delivery, and evaluation. Educators and assessment designers may make educated judgments about how to employ

AI-based assessment generators to enhance learning and assessment by knowing their benefits and limits.

2.5 *Impact of AI Technology on Assessment Design and Implementation*

AI technology has had a significant impact on assessment design and implementation in recent years, leading to the development of new assessment models and strategies.

The creation of adaptive evaluation models has had a significant influence on AI technology [29, 30]. These models employ machine learning algorithms to examine learner data and provide evaluations that are personalized to the requirements and abilities of individual learners. Adaptive assessments have been found to be more successful in promoting learning and improving results than standard evaluations [31–34].

AI technology has also enabled the development of formative assessment tools that provide immediate feedback to learners [35, 36]. These tools use natural language processing and other AI techniques to analyze learner responses and provide targeted feedback in real time. Formative assessment tools have been shown to be effective at promoting learning and engagement [37].

In addition to these new assessment models and strategies, AI technology has also had a significant impact on assessment implementation. AI algorithms can analyze large amounts of data quickly and accurately, which can reduce the time and resources required for assessment design and delivery [2, 38]. AI algorithms can also analyze assessment data to identify patterns and trends, which can help educators improve assessment design and identify areas where learners may need additional support.

The influence of AI technology on assessment and implementation, on the other hand, is not without issues. The requirement to ensure the validity and reliability of AI-based evaluations is a key problem [39, 40]. AI algorithms may not be able to capture the entire spectrum of information, skills, and talents being evaluated, resulting in incomplete or erroneous evaluations.

Despite these obstacles, AI technology has had a considerable influence on assessment design and execution, and educators and assessment designers are likely to continue looking for new methods to use AI to improve assessment outcomes.

2.6 *Ethical and Social Considerations in the Use of AI in Assessment Generators*

The ethical and social implications of using AI in assessment generators should be carefully considered. One major concern is the potential, for bias in AI-driven assessments [12]. It is possible that AI algorithms can perpetuate existing biases in

the data they are trained on resulting in evaluations of learners from underrepresented groups [41].

Another ethical consideration involves ensuring transparency and explainability in AI-based assessments [42]. Learners and educators should understand how AI algorithms make decisions regarding assessment outcomes. They should also have the ability to question or contest decisions that appear unjust or inaccurate.

Privacy and data security are concerns raised by the use of AI in assessment generators [43, 44]. Large amounts of learner data, which may include information like details or learning disabilities might be needed to train AI algorithms. Educators and assessment designers must take measures to securely store learner data and ensure it is only used for its intended purpose.

Furthermore, incorporating AI into assessment generators presents issues related to the role of technology in education [45]. There is a possibility that AI-based evaluations could be seen as replacing forms of assessment potentially undermining judgment and expertise, in education.

It is extremely important to consider the cultural implications of using AI in assessment generators. We need to ensure that technology is used in a way that supports the goals of hindering them.

In summary, the ethical and societal concerns raised by the implementation of intelligence in assessment generators are complex and multifaceted. It is crucial for educators and assessment designers to be mindful of these challenges and make every effort to create AI-based exams that are fair, transparent, and respectful of learner privacy and autonomy.

3 Methodology

A systematic review approach was meticulously employed to conduct an extensive analysis of the body of literature concerning the intersection of artificial intelligence and assessment generators within the domain of education. This methodological framework was structured to encompass the formulation of inclusion and exclusion criteria, the identification of pertinent databases, an exhaustive literature search, and the subsequent extraction and analysis of data derived from selected research studies.

To commence, a set of rigorous inclusion and exclusion criteria was meticulously crafted to ensure the utmost relevance of the chosen papers to the core research question and objectives. Inclusion criteria stipulated that the selected papers must investigate the application of artificial intelligence in education assessment generators, be published in the English language, and originate from peer-reviewed publications. Conversely, papers deviating from the scope of the research, those published in non-peer-reviewed journals, or those lacking full-text availability were systematically excluded.

Following the criteria definition, a comprehensive exploration of relevant resources was undertaken, drawing from esteemed scholarly databases such as Web of Science, ERIC, and Scopus. A comprehensive set of search terms was systematically

utilized, including terms such as “Artificial intelligence,” “Assessment generators,” “Education,” “Machine learning,” “Computer-based assessment,” and “Automated assessment.“ The initial search yielded a total of 986 studies. Subsequently, duplicate studies were meticulously eliminated through a thorough examination of titles and abstracts, resulting in the refinement of the dataset.

Ultimately, 87 papers were chosen for an exhaustive full-text examination following this initial screening phase. These selected studies were subjected to a meticulous review to ascertain their alignment with the inclusion criteria. The literature evaluation ultimately encapsulated a total of 32 papers.

The subsequent phase of the methodology encompassed a systematic data extraction and analysis process. The extracted data encompassed key information pertaining to the study design, sample size, and educational level, the specific AI and assessment generator technologies employed, and the principal findings and conclusions of each study. The data underwent a rigorous analytical procedure aimed at identifying underlying patterns and themes directly related to the research question and objectives.

4 Results and Discussion

4.1 Overview of the Literature Review Findings

The systematic review found 32 articles that matched the inclusion criteria and offered useful information on the usage of AI and assessment generators in education. The selected studies were published between 2014 and 2022 and addressed a wide variety of educational levels, including elementary, secondary, and higher education. The studies employed various types of machine learning algorithms, with neural networks being the most frequently used (12 studies). Table 1 and Fig. 5 summarize the number of studies that used each type of machine learning algorithm.

The studies were further divided into levels of education, with six concentrating on basic education, eight on secondary education, and 18 on higher education. The

Table 1 Studies by type of machine learning used

Type of machine learning	Number of studies
Neural networks	12
Decision trees	5
Bayesian networks	3
Support vector machines	2
Fuzzy logic	2
Others	8
Total	32

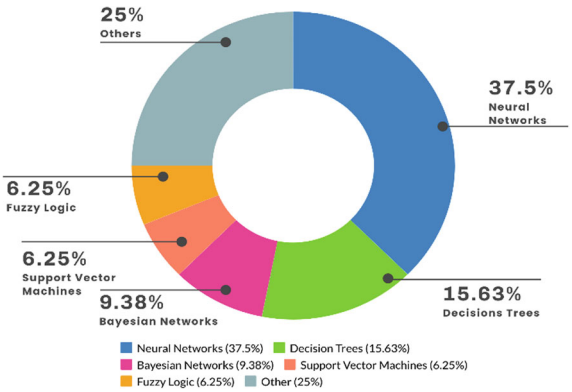


Fig. 5 Studies by type of machine learning used

amount of research that focused on each educational level is summarized in Table 2 and Fig. 6.

The literature analysis reveals the advantages of AI and assessment generators, including improved efficiency, impartiality, and personalized feedback for students. However, challenges such as fairness, bias, and privacy concerns are significant drawbacks. The application of AI in education for assessment purposes is still in its nascent phase, necessitating further research to explore its potential to enhance assessment quality and student learning outcomes. Addressing ethical and social issues is imperative to ensure the fair, transparent, and responsible utilization of AI in educational evaluations.

Table 2 Studies by educational level

Education level	Number of studies
Primary	12
Secondary	5
Higher education	3
Total	32

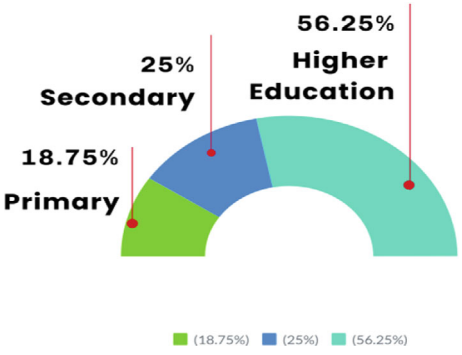


Fig. 6 Studies by educational level

5 Summary of the Benefits and Limitations of AI and Assessment Generators in Education

The literature review identified several benefits and limitations of AI and assessment generators in education. Table 3 summarizes the main advantages and limitations of this technology.

Based on research findings the implementation of AI and assessment generators has shown potential in enhancing the efficiency and fairness of assessments, through automated grading and prompt feedback provision for students. This advanced technology enables the delivery of feedback and adaptive exams tailored to meet students' needs.

However, it is crucial to acknowledge limitations when utilizing AI in evaluation processes. One significant concern revolves around the potential for bias and unfairness in assessments particularly if algorithms are not appropriately calibrated or trained on datasets. Additionally ensuring valid AI generated assessments requires the implementation of quality assurance techniques.

Another aspect to consider is the absence of interaction throughout the evaluation process, which may impact student motivation and engagement negatively. Furthermore, the intricacies associated with AI technology and assessment generators might pose challenges towards achieving acceptance and implementation.

Addressing privacy and security concerns is also paramount to ensure the handling and utilization of student data. The integration of intelligence in evaluations raises questions regarding data ownership, consent, and transparency that necessitate careful exploration and management.

5.1 Analysis of Studies by Type of Machine Learning and Educational Level

The studies were graded depending on the type of machine learning used and the educational level at which they were completed. Table 4 summarizes the findings of this inquiry.

The analysis discovered that the use of AI and assessment generators varied depending on the level and type of machine learning. Across all levels of education

Table 3 Advantages and limitations of AI and assessment generators

Advantages	Limitations
Increased efficiency	Concerns about fairness and bias
Objectivity	Need for quality assurance
Personalized feedback	Lack of human interaction
Adaptive assessment	Technical complexity
Immediate feedback	Privacy and security concerns

Table 4 Analysis of studies by type of machine learning and educational level

Type of machine learning	Primary	Secondary	Higher education
Neural networks	4	4	4
Decision trees	1	3	1
Bayesian networks	0	2	1
Support vector machines	1	1	0
Fuzzy logic	0	1	1
Others	1	1	6
Total	13	12	13

neural networks were the employed type of machine learning. However, decision trees and Bayesian networks were also popular, in higher education.

In school there was a focus on utilizing decision trees and support vector machines. While neural networks and decision trees were widely used in school higher education saw a range of machine learning techniques being employed including neural networks, Bayesian networks and fuzzy logic.

Research conducted in schools mainly concentrated on assessing skills such as reading comprehension and problem solving in math. In schools research expanded to explore the application of AI and assessment generators across subjects like language acquisition, science and social studies. Research efforts in education delved into themes such, as evaluating critical thinking skills promoting collaborative learning practices and incorporating AI into assessment design.

5.2 Comparison of AI and Assessment Generators to Traditional Assessment Methods

In this subsection, we will compare the use of AI powered assessment tools, with conventional assessment methods. After reviewing research, it became apparent that AI and assessment generators provide benefits, over traditional assessment methods. However, it is important to acknowledge that they also come with their limitations that need to be considered.

As shown in Table 5 assessment generators powered by AI offer advantages compared to assessment methods.

One of the benefits is their efficiency as these generators can quickly and effortlessly produce evaluations saving an amount of time. Additionally, AI based evaluation generators can be customized to cater to student needs helping students better comprehend the information presented.

Another advantage of AI-based assessment generators is the consistency in grading. These generators assess assignments using criteria that reduce subjectivity in the grading process, ensuring that students are evaluated consistently and fairly.

Table 5 Comparison of AI-based assessment generators to traditional assessment methods

Factors	AI-based assessment generators	Traditional assessment methods
Efficiency	Can generate assessments quickly and easily	Time-consuming process
Customization	Can be customized to meet individual student needs	Limited flexibility
Consistency	Consistent and objective grading	Subjective grading
Adaptability	Can adapt to changes in student needs and abilities	Less adaptable
Accuracy	Can provide accurate assessment data	Limited accuracy

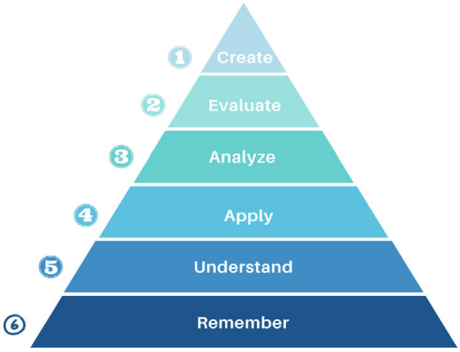


Fig. 7 Bloom's taxonomy revised by Anderson and Krathwohl [46]

However, it is important to consider some limitations when using AI-based evaluation generators. For instance, these generators may have limitations in terms of adaptability, making it challenging to tailor assessments for individual learners. They may also struggle with adjusting examinations based on changes in student needs and skills. Lastly, accuracy plays a significant role. While AI-based assessment generators can provide valuable assessment data, traditional assessment methods may prove more accurate in certain cases, particularly when evaluating higher-level cognitive skills (see Fig. 7).

5.3 Ethical and Social Considerations in the Use of AI in Assessment Generators

The literature review has unveiled a set of ethical and social concerns that demand thorough assessment and effective measures when integrating AI into assessment generators. In Table 5, these factors are succinctly summarized, along with corresponding recommendations and best practices to mitigate these concerns.

The literature study underscores the importance of addressing these social concerns when implementing AI in assessment generators. To ensure fairness, privacy, transparency, and pedagogical validity, a set of recommended best practices has been presented in Table 6. Furthermore, recognizing the value of human interaction and student engagement in the assessment process remains vital, alongside the responsible use of AI to enhance educational practices.

6 Conclusion and Future Work

In summary, our review comprehensively explored the intersection of Artificial Intelligence and assessment generators within the realm of education. Through a meticulous analysis of 32 studies, we delineated the evolution of AI and assessment generators, delved into underlying frameworks, and evaluated advantages and limitations. Our results underline the potential of AI-based assessment generators in improving efficiency, imparting impartial feedback, and boosting learning outcomes. However, we identified concerns such as biases and lack of transparency, emphasizing the need for rigorous standards and pedagogical integrity in assessments.

Addressing the research questions, our study underscores the transformative promise of AI in reforming educational assessments. The integration of AI offers opportunities for enhanced educational outcomes, but the delicate balance between promise and peril must be maintained. Practical implementation demands meticulous attention to privacy, security, biases, and discrimination.

As for the future, it is imperative to approach AI integration in education with caution, implementing necessary safeguards. Future research endeavors should focus on unraveling the nuanced impacts of AI, while continuing to explore the ethical and pedagogical dimensions of this technology. By doing so, we can harness the full potential of AI-based assessment generators, ensuring they truly enrich educational experiences and outcomes.

Finally, AI-based assessment generators open up interesting new avenues for enhancing educational outcomes and experiences. To fully appreciate these benefits, however, we must continue to critically assess and examine the technology's ethical and pedagogical implications. In addition several methods [47–63] could be adapted to this area to enrich the scientific literature and extract new ideas for their use in other vital areas.

Table 6 Ethical and social considerations in the use of AI in assessment generators

Considerations	Explanation	Recommendations and best practices
Fairness and bias	AI algorithms can perpetuate existing biases and unfairness in assessment if not properly calibrated or trained on diverse data sets	• Regularly audit and retrain AI algorithms to reduce bias • Ensure diverse and representative training datasets • Implement fairness-aware algorithms to mitigate bias
Privacy and security	The use of AI in assessment raises concerns about protecting student data and the potential for privacy and security breaches	• Comply with data protection regulations • Anonymize and secure student data to prevent breaches • Implement strong encryption and access controls
Transparency and accountability	There is a need for transparency and accountability in the use of AI in assessment, to ensure that the algorithms are reliable, valid, and ethical	• Document AI assessment processes and decisions for transparency • Develop clear guidelines for accountability in AI-driven assessments • Implement third-party audits to ensure ethical compliance

(continued)

Table 6 (continued)

Considerations	Explanation	Recommendations and best practices
Human interaction	The lack of human interaction in the assessment process can have negative effects on student motivation and engagement and can lead to a lack of understanding and trust in the assessment process	• Incorporate opportunities for human interaction, such as instructor feedback • Design assessments that strike a balance between AI-driven and human involvement • Ensure clear communication about the role of AI in assessments
Pedagogical validity	The use of AI in assessment should be aligned with pedagogical principles and should not replace human judgment in the assessment process	• Collaborate with educators to align AI-driven assessments with pedagogical goals • Use AI to enhance, not replace, human judgment in assessments • Continuously evaluate the pedagogical impact of AI in assessment

References

1. Wang, P.: On defining artificial intelligence. J. Artif. Gen. Intell. **10**, 1–37 (2019)
2. Zhai, X., et al.: A review of artificial intelligence (AI) in education from 2010 to 2020. Complexity **2021**, 1–18 (2021)
3. Hernández-Orallo, J.: Evaluation in artificial intelligence: from task-oriented to ability-oriented measurement. Artif. Intell. Rev. **48**, 397–447 (2017)
4. Aldowah, H., Al-Samarraie, H., Fauzy, W.M.: Educational data mining and learning analytics for 21st century higher education: a review and synthesis. Telemat. Inform. **37**, 13–49 (2019)
5. Lee, I., Ali, S., Zhang, H., DiPaola, D., Breazeal, C.: Developing middle school students' AI literacy. In: Proceedings of the 52nd ACM Technical Symposium on Computer Science Education, pp. 191–197 (2021)
6. Holmes, W., et al.: Ethics of AI in education: towards a community-wide framework. Int. J. Artif. Intell. Educ. 1–23 (2021)
7. Ertmer, P.A., Ottenbreit-Leftwich, A.T.: Emerging technologies in education: promise and potential peril. Handbook
8. Rupp, A.A., Templin, J., Henson, R.A.: Diagnostic Measurement: Theory, Methods, and Applications. Guilford Press (2010)
9. Leighton, J., Gierl, M.: Cognitive Diagnostic Assessment for Education: Theory and Applications. Cambridge University Press (2007)

10. Wang, F., et al.: Neural cognitive diagnosis for intelligent education systems. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, pp. 6153–6161 (2020)
11. Paulsen, J., Valdivia, D.S.: Examining cognitive diagnostic modeling in classroom assessment conditions. J. Exp. Educ. **90**, 916–933 (2022)
12. Baker, R.S., Martin, T., Rossi, L.M.: Educational data mining and learning analytics. Wiley Handb. Cogn. Assess. Framew. Methodol. Appl. 379–396 (2016)
13. Muñoz, J.L.R., et al.: Systematic review of adaptive learning technology for learning in higher education. Eurasian J. Educ. Res. **98**, 221–233 (2022)
14. El-Sabagh, H.A.: Adaptive e-learning environment based on learning styles and its impact on development students' engagement. Int. J. Educ. Technol. High. Educ. **18**, 1–24 (2021)
15. Peng, H., Ma, S., Spector, J.M.: Personalized adaptive learning: an emerging pedagogical approach enabled by a smart learning environment. Smart Learn. Environ. **6**, 1–14 (2019)
16. VanLehn, K., Siler, S., Murray, C., Yamauchi, T., Baggett, W.B.: Why do only some events cause learning during human tutoring? Understanding the tutoring effectiveness hypothesis. Cogn. Instr. **23**, 293–329 (2005)
17. Bienkowski, M., Feng, M., Means, B.: Enhancing teaching and learning through educational data mining and learning analytics: an issue brief. Off. Educ. Technol. US Dep. Educ. (2012)
18. Albert, D., Lukas, J.: Knowledge Spaces: Theories, Empirical Research, and Applications. Psychology Press (1999)
19. Abdelrahman, G., Wang, Q., Nunes, B.: Knowledge tracing: a survey. ACM Comput. Surv. **55**, 1–37 (2023)
20. Almond, R.G., Mislevy, R.J., Steinberg, L.S., Yan, D., Williamson, D.M.: Bayesian Networks in Educational Assessment. Springer (2015)
21. Vomlel, J.: Bayesian networks in educational testing. Int. J. Uncertain. Fuzziness Knowl.-Based Syst. **12**, 83–100 (2004)
22. Waldmann, M.R., Martignon, L.: A Bayesian network model of causal learning. In: Proceedings of the Twentieth Annual Conference of the Cognitive Science Society, pp. 1102–1107. Routledge (2022)
23. Kitson, N.K., Constantinou, A.C., Guo, Z., Liu, Y., Chobtham, K.: A survey of Bayesian network structure learning. Artif. Intell. Rev. 1–94 (2023)
24. Gordon, A.D., Henzinger, T.A., Nori, A.V., Rajamani, S.K.: Probabilistic programming. In: Future of Software Engineering Proceedings, pp. 167–181 (2014)
25. Brusilovsky, P., Millan, E.: User models for adaptive hypermedia and adaptive educational systems. In: The Adaptive Web: Methods and Strategies of Web Personalization, pp. 3–53. Springer (2007)
26. Vincent-Lancrin, S., Van der Vlies, R.: Trustworthy artificial intelligence (AI) in education: promises and challenges (2020)
27. Huang, Y., Khan, S.M.: Advances in AI and machine learning for education research. Comput. Psychom. New Methodol. New Gener. Digit. Learn. Assess. Ex. R Python 195–208 (2021)
28. Eysenbach, G., et al.: The role of ChatGPT, generative language models, and artificial intelligence in medical education: a conversation with ChatGPT and a call for papers. JMIR Med. Educ. **9**, e46885 (2023)
29. Conejo, R., et al.: SIETTE: a web-based tool for adaptive testing. Int. J. Artif. Intell. Educ. **14**, 29–61 (2004)
30. Chen, X., Zou, D., Xie, H., Cheng, G., Liu, C.: Two decades of artificial intelligence in education. Educ. Technol. Soc. **25**, 28–47 (2022)
31. Ribeiro, M.T., Lundberg, S.: Adaptive testing and debugging of NLP models. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Vol. 1: Long Papers), pp. 3253–3267 (2022)
32. Ma, W., Adesope, O.O., Nesbit, J.C., Liu, Q.: Intelligent tutoring systems and learning outcomes: a meta-analysis. J. Educ. Psychol. **106**, 901 (2014)
33. Fang, Y., Ren, Z., Hu, X., Graesser, A.C.: A meta-analysis of the effectiveness of ALEKS on learning. Educ. Psychol. **39**, 1278–1292 (2019)

34. Steenbergen-Hu, S., Cooper, H.: A meta-analysis of the effectiveness of intelligent tutoring systems on college students' academic learning. J. Educ. Psychol. **106**, 331 (2014)
35. Li, T., Reigh, E., He, P., Adah Miller, E.: Can we and should we use artificial intelligence for formative assessment in science? J. Res. Sci. Teach. (2023)
36. Gonzalez-Calatayud, V., Prendes-Espinosa, P., Roig-Vila, R.: Artificial intelligence for student assessment: a systematic review. Appl. Sci. **11**, 5467 (2021)
37. Roschelle, J.: Intelligence augmentation for collaborative learning. In: International Conference on Human-Computer Interaction, pp. 254–264. Springer (2021)
38. Owan, V.J., Abang, K.B., Idika, D.O., Etta, E.O., Bassey, B.A.: Exploring the potential of artificial intelligence tools in educational measurement and assessment. Eurasia J. Math. Sci. Technol. Educ. **19**, em2307 (2023)
39. Brusilovsky, P.: Adaptive educational hypermedia. Int. PEG Conf. **10**, 8–12 (2001)
40. Mulwa, C., Lawless, S., Sharp, M., Arnedillo-Sanchez, I., Wade, V.: Adaptive educational hypermedia systems in technology enhanced learning: a literature review. In; Proceedings of the 2010 ACM Conference on Information Technology Education, pp. 73–84 (2010)
41. Buolamwini, J., Gebru, T.: Gender shades: Intersectional accuracy disparities in commercial gender classification. In: Conference on Fairness, Accountability and Transparency, pp. 77–91. PMLR (2018)
42. Lipton, Z.C.: The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. Queue **16**, 31–57 (2018)
43. Case, C.J., King, D.L., Case, J.A.: E-cheating and undergraduate business students: trends and the role of gender. J. Bus. Behav. Sci. **31**, 102–113 (2019)
44. Nigam, A., Pasricha, R., Singh, T., Churi, P.: A systematic review on AI-based proctoring systems: past, present and future. Educ. Inf. Technol. **26**, 6421–6445 (2021)
45. Selwyn, N.: Minding our language: why education and technology is full of bullshit… and what might be done about it. Learn. Media Technol. **41**, 437–443 (2016)
46. Krathwohl, D.R.: A revision of Bloom's taxonomy: an overview. Theory Pract. **41**, 212–218 (2002)
47. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks. ArXiv Prepr. ArXiv13075910 (2012)
48. Boutyour, Y., Idrissi, A.: Adaptive decentralized policies with attention for large-scale multi-agent environments. IEEE Trans. Artif. Intell. 1–10 (2024) https://doi.org/10.1109/TAI.2024.3415550
49. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. Int. Conf. Big Data Adv. Wirel. Technol. (2016)
50. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF 107–116 (2006)
51. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
52. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
53. Abourezq, M., Idrissi, A.: A Cloud Services Research and Selection System. IEEE ICMCS (2014)
54. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. Proc. Int. Conf. Internet Things Cloud Comput. (2016)
55. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
56. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners-topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. 5567–5584 (2023)
57. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)

58. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
59. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
60. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2021) (2020)
61. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. Proc. Int. Conf. Big Data Adv. Wirel. Technol. (2016)
62. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
63. Boutyour, Y., Idrissi, A.: Deep reinforcement learning in financial markets context: review and open challenges. In: Studies in Computational Intelligence, pp. 49–66. Springer Nature Switzerland (2023). https://doi.org/10.1007/978-3-031-33309-5_5

Personalized Course Recommender System Based on Multiple Approaches: A Comparative Analysis

Hajar Majjate, Youssra Bellarhmouch, Adil Jeghal, Ali Yahyaouy, Hamid Tairi, and Khalid Alaoui Zidani

Abstract Nowadays, online learning platforms offer an excellent opportunity for learners to access various courses from various fields, breaking down geographical barriers and promoting lifelong learning. However, it can be overwhelming for students to select the appropriate course among the numerous options available; it is also exhausting for a learner to find similar courses developing the same characteristics and skills as an already taken course. Thus, recommendation systems are crucial in helping online learners filter suitable content and make the right decision during course selection, which positively impacts student engagement and the overall learning experience. However, many widely used e-learning platforms have yet to incorporate recommendation engines. Additionally, educational platforms do not benefit from a sophisticated and accurate recommendation system like those found on streaming services, social media platforms, and online marketplaces. This paper presents a personalized course recommender system based on multiple recommendation approaches, including collaborative filtering, content-based filtering, popularity-based models, and hybrid models. The system is designed to aid online learners in selecting courses that match their interests with personalized content. It offers solutions for new users by suggesting popular courses and other users' preferences. To ensure precision and evaluate the efficiency of each model, we evaluated their performance using a range of metrics, such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Precision, and Recall, in a comparative context.

Keywords Collaborative filtering · Content-based filtering · Popularity-based models · Hybrid filtering · E-learning · Recommendation system · Personalized learning

H. Majjate (✉) · Y. Bellarhmouch · A. Yahyaouy · H. Tairi · K. A. Zidani
LISAC Laboratory, Faculty Of Science Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, 30030 Fez, Morocco
e-mail: hajar.majjate@usmba.ac.ma

A. Jeghal
National School of Applied Sciences Fes (ENSA), Sidi Mohamed Ben Abdellah University, 30030 Fez, Morocco

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_22

1 Introduction

With the growth of the online community, many companies implement recommendation systems on their websites to increase their revenues because the key to the success of any business is understanding customers' preferences and interests. Recommendation systems can be defined as a subclass of information filtering mechanisms that aim to predict user reaction [1, 36, 37] towards a specific item [1, 22], be it a product, service, or streaming video; this user's reaction may be reflected in a rating, a purchase, or the amount of time spent engaging with the item, providing insight into their preferences. Recommendation systems explore data [2] provided by users to suggest items that may interest them. This process is generated through various statistical techniques and machine-learning solutions.

Recommendation systems have proven to be an invaluable tool for filtering information in various fields. In the educational area, which has also been broadly digitized, online learning is increasingly being adopted, particularly after the pandemic period, which highlighted the critical role of e-learning platforms as the best alternative environment for learning, promoting student success and skill development, accessible to learners of all backgrounds from anywhere around the globe, with an unparalleled opportunity to expand their knowledge, through a large variety of courses and educational contents in various domains, making e-learning even more valuable than traditional methods.

However, this vast amount of information available [3, 4] can also overwhelm learners, causing information and content overload [4], disorientation, and wasting time searching for suitable courses. To address this challenge, recommender systems are increasingly implemented in e-learning platforms to offer learners personalized service and relevant educational content.

Despite their significance, it's worth noting that the application of recommender systems engines in the educational field still needs to be improved compared to other domains such as e-commerce, social networks, entertainment, and streaming websites [5].

The main contribution of this study is:

- developing a personalized e-learning course recommendation system in a learner-friendly context that provides highly accurate recommendations, considering various user profiles, even the newly registered ones. To achieve this, we incorporated multiple filtering approaches, including Item-collaborative, Content-Based, Popularity-based, and Hybrid-Based Filtering.
- The evaluation of the performance of each model is presented in a comparative context using various metrics, such as the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), Precision, and Recall, in order to identify which type of recommendation model performs well for educational data.
- The study involves statistical measures commonly used in streaming websites to generate accurate recommendations, such as the weighted rating formula broadly used in movie recommendations.

- The objective of this research is to protect learner privacy by avoiding the exploitation of their historical reactions, instead focusing solely on public reactions such as the course rating.
- This work aims to overcome the limitation of recommender systems that rely only on popularity-based recommendations by adopting a hybrid approach.

This paper is structured into five distinct sections. In Sect. 2, the related works are discussed, along with an insightful overview of the primary recommendation system approaches. Section 3 articulates the methodology, including data analysis and a comprehensive description of the proposed architecture. Section 4 is dedicated to presenting the model evaluation results, while Sect. 5 concludes the work.

2 Related Works

Recommender Systems have become an essential area in academic literature due to their importance in improving many research fields. Our focus in this study is specifically on their application in the educational field. Several research studies have been conducted proposing various recommendation approaches for the educational sector. This section briefly overviews past research on integrating recommender systems into e-learning platforms.

Walker et al. (2004) [6] proposed the Altered Vista (AV) system, which is recognized as one of the earliest examples of collaborative filtering systems designed to recommend learning resources. It relied on gathering feedback from learners through comments and evaluations concerning the learning content, which were then shared to provide word-of-mouth recommendations.

Klašnja-Milićević et al. (2015) [7] presented an overview of many challenges to designing a recommender system in e-learning environments. They have introduced various limitations of the past few years' generations of recommendation techniques and possible extensions with a model for tagging activities and tag-based recommender systems on e-learning environments.

Tan et al. (2008) [8] Proposed an online e-learning recommendation system requirement analysis based on a user-based collaborative filtering approach.

Bobadilla et al. (2013) [9] overview the evolution of recommender systems and collaborative filtering methods over 253 research papers, including future possibilities in the age of the Internet of Things.

Zhang et al. (2021) [10] reviewed the main recommendation techniques employed in the E-learning environment. The study emphasizes the importance of recommender systems in improving higher education and highlights potential future directions for these systems in E-learning.

Gulzar et al. (2018) [11] presented a personalized e-learning recommendation system based on a hybrid approach and ontology to suggest and guide learners in selecting the courses per their requirements to increase the effectiveness of online learning.

Youness et al. (2019) [12] proposed a course recommendation system based on social filtering and collaborative filtering approaches, especially on past learning and rating information, to recommend relevant and personalized course content that suits every learner's profile.

Morsomme and Alferez (2019) [13] proposed a content-based recommendation system for Liberal Arts students at Maastricht University of Netherlands. The goal was to help students select appropriate courses that considered their interests and academic expertise. The recommendation system is based on a topic model that uses course descriptions, and it also includes a predictive model for grades based on students' past academic performance and level of expertise.

In this research, we propose an adaptable recommendation framework that overcomes the limitations of conventional content-based and item-based collaborative filtering approaches. Our model combines popularity-based recommendations with both content-based and item-based collaborative filtering approaches, resulting in a personalized and diverse range of recommendations for students to enhance their course selection in online learning. We also evaluate the accuracy of each recommendation method to ensure the best outcomes for students.

3 Recommendation Filtering Techniques

Generally, Recommender systems techniques are classified into three main categories [14, 15]: collaborative recommender systems [16], content-based recommender systems [17], and hybrid recommender systems [18].

Content-Based recommendation: the content-based filtering system recommends items to users that are similar to the ones they have liked in the past. The principle of the process is performed by matching user preferences with item attributes [19], and it analyses item descriptions [20] to make predictions based on similar features of the item. [21]

Content-based filtering doesn't waste time collecting information about other users; [21] it collects only results of previous user research in order to recommend relevant items in harmony with his specific preferences.

Collaborative Recommender Systems: Collaborative recommender systems are the most used technique for information filtering [22]. It recommends items to users based on what other users with similar tastes have liked previously [23]. The collaborative recommender methods are commonly divided into neighborhood-based and model-based approaches [22, 24, 25].

Neighbourhood-based and model-based approaches produce predictions for users based on their ratings. Neighborhood-based methods rely on selecting a group of users with similar preferences, while model-based methods create predictive models based on the ratings of all users [22–24].

Hybrid recommender systems: despite their success, the content-based filtering techniques and collaborative approaches have presented several limitations which

Table 1 Variables

Variable name	Variable type
Course_title	Categorical
Organisation_title	Categorical
Course_description	Categorical
Course_duration	Categorical
Weekly_hours	Categorical
Course_type	Categorical
Course_rating	Numerical
Course_reviews	Numerical

impacted negatively the accuracy and the reverence of the presentation recommendation that offers each model [23], such as limited diversity of content to analysis, Feature Extraction problems, data sparsity [23, 26]and overspecialization [23, 27] related to the content-based filtering approaches, and Cold Start problem[26], data privacy issues and more related to the collaborative filtering approaches.

A hybrid filtering system has been proposed [23, 28, 29] to overcome the limitations of using only content-based or collaborative filtering techniques. This system combines the strengths of both approaches. It allows the inclusion of additional methods, such as association rules [30], knowledge-based [31], context-aware recommendations [32], and more, to improve recommendation performance and reach user satisfaction.

4 Methods

4.1 Data Description

The dataset provides information on 1,599 online courses, each with 8 different variables. These variables include categorical descriptions such as the course title, the name of the organization offering the course, a description of the course content, course duration, expected weekly hours of study, and course type. In addition to these, two numerical variables are provided: the course rating, which shows student evaluations, and the number of reviews submitted for each course. Table 1 provides further details regarding the data variables.

4.2 Proposed System

The objective of the proposed architecture is to recommend courses to learners on an e-learning platform using a combination of multiple approaches in order to generate

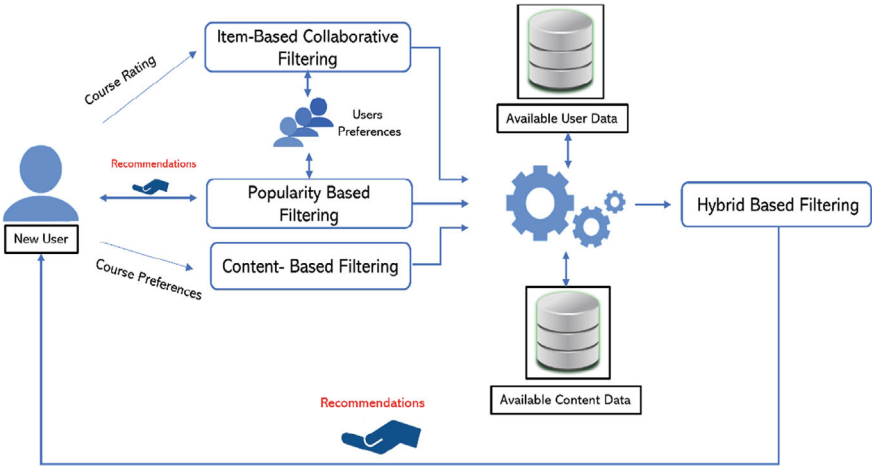


Fig. 1 Proposed system

relevant and helpful recommendations. This aims to overcome limitations related to traditional recommendation engines, such as the cold start problem.

To address this issue, we proposed, in the first level, as shown in Fig. 1, recommending the most popular courses to newly enrolled users based on the ratings given by other learners. This enables them to explore and discover new courses and subjects through the e-learning platform.

In the popularity-based filtering, we adopted the Internet Movie Database (IMDB) [33] weighted rating methods, commonly used in streaming platforms, to sort courses in descending order based on Course_reviews and Course_rating.

(IMDb)'s formula intitules weighted rating (WR) given as follows:

$$\mathrm{WR} = \left(\frac{\mathrm{v}}{v+m} \cdot R\right) + \left(\frac{m}{v+m} \cdot C\right) \tag{1}$$

Were,

- v is the number of votes for each course.
- m is the minimum votes required to be listed in the chart.
- R is the average rating of the course.
- C is the mean rate across the whole report.

In the second stage, the system gathers information from new learners based on the course title and description they search. This data predicts their future preferences and recommends courses that match their interests.

The recommendation system uses a content-based filtering approach. It converts textual data into numerical vectors using Term Frequency-Inverse Document Frequency (TF-IDF) Vectorization [34]. Cosine similarity [35] is then applied to

measure the similarity between these vectors. The system does not collect personal data or consult the user's information or profile.

Insufficient data on the learner's preferred content prompts the system to switch to collaborative filtering. The Item-Based Collaborative Filtering method calculates cosine similarity between courses based on 'Course_rating' and 'Course_reviews.' It recommends new courses based on users' and other users' past ratings and reviews, indicating their interests, knowing that the two selected items are public information that implicitly indicates users' tastes and interests.

The hybrid mode is an advanced feature that automatically switches between all the recommendation methods on the platform based on available input. It provides different types of recommendations based on the importance given to each recommendation method. The switching is done based on a future analysis of available data to generate the most suitable recommendation.

The hybridization of the recommendation is designed to be user-friendly and respects the privacy and content of the users by only collecting public data. Additionally, it overcomes the cold start problem by helping learners discover new courses and ensure the continuity of learning.

5 Results and Discussion

To evaluate the performance of the proposed models, we have selected multiple evaluation metrics: the Root Mean Squared Error (RMSE), The Mean Absolute Error (MAE), and the Precision and Recall. These metrics are further detailed in Fig. 2.

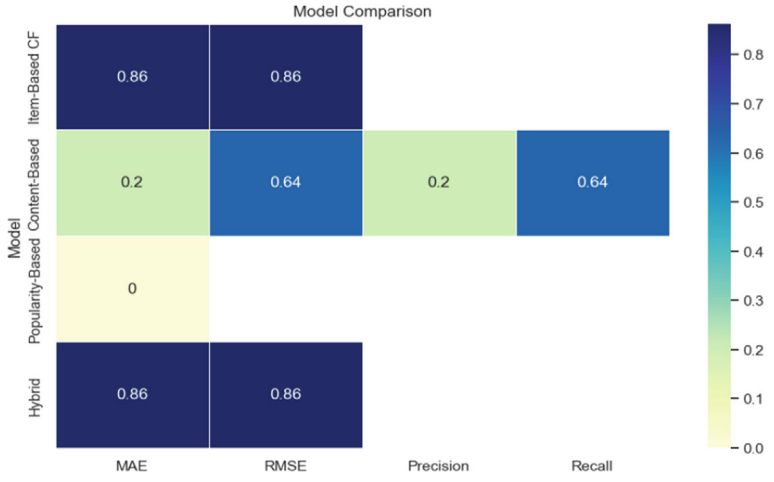


Fig. 2 Models evaluation

The heatmap displays the evaluation metrics (MAE, RMSE, Precision, and Recall) for each recommendation model. Each cell in the heatmap represents a specific metric value for a particular model. The color intensity indicates the relative value of the metric.

Overall, based on the obtained results:

- The Content-Based model is impressively the best-performing model, as it has the lowest MAE, RMSE, and good Precision and Recall values. It provides accurate recommendations based on course content.
- The hybrid model performs well but has slightly higher MAE and RMSE than the content-based model. However, it offers different recommendation solutions.
- The Popularity-Based model scores a "0" for both MAE and RMSE, as these metrics do not apply to its non-personalized recommendation approach. This model suggests popular or highly rated items to all users rather than personalized predictions. As a result, there are no individualized rating predictions to calculate errors. Given its reliance on item popularity, evaluating the performance of this model using MAE and RMSE is not appropriate, as these metrics are designed for models providing personalized predictions.
- The Item-Based CF model has the highest MAE and RMSE among the evaluated models, indicating it needs to be more accurate in providing recommendations.

Combining multiple recommendation approaches can make the recommendation model more powerful and personalized. Each model has limitations that can be improved by adding the benefits of other models. This helps to provide more relevant and personalized recommendations for every learner profile while respecting their privacy and personal data.

6 Conclusion

In this paper, we have focused on developing personalized course recommender systems that use multiple filtering approaches. We aim to implement this system on an e-learning platform, recognizing that the education sector deserves to be a part of the growth of the internet and the emergence of the artificial intelligence field. Providing learners a customized and engaging learning experience is crucial for their academic success and performance. E-learning platforms are the future of learning and should be a space that encourages learning new things, promotes lifelong learning, and provides suitable opportunities for people to improve their careers and develop their skills.

In future work, we plan to enhance this model with other trending recommendation approaches based on different criteria while respecting learners' privacy and personal information.

References

1. Sharma, M., Mann, S.: A survey of recommender systems: approaches and limitations. Int. J. Innov. Eng. Technol. **2**(2), 8–14 (2013)
2. Wang, S., Wang, Y., Sivrikaya, F., et al.: Data science for next-generation recommender systems. Int. J. Data Sci. Anal. **16**, 135–145 (2023)
3. Jallouli, M., Lajmi, S., Amous, I.: Designing recommender system: conceptual framework and practical implementation. Procedia Comput. Sci. **112**, 1701–1710 (2017)
4. Patel, K., Patel, H.: A state-of-the-art survey on recommendation system and prospective extensions. Comput. Electron. Agric. **178**, 105779 (2020)
5. Christensen, I.A., Schiaffino, S.: Entertainment recommender systems for group of users. Expert Syst. Appl. (2011). https://doi.org/10.1016/j.eswa.2011.04.221
6. Walker, A., et al.: Collaborative information filtering: a review and an educational application. Int. J. Artif. Intell. Educ. **14**, 1–26 (2004)
7. Klašnja-Milićević, A., Ivanović, M., Nanopoulos, A.: Recommender systems in e-learning environments: a survey of the state-of-the-art and possible extensions. Artif. Intell. Rev. **44**(4), 571–604 (2015). https://doi.org/10.1007/s10462-015-9440-z
8. Tan, H., Guo, J., Li, Y.: E-learning recommendation system. In: Proceedings of the International Conference on Computer Science and Software Engineering, Wuhan, China, pp. 430–433 (2008)
9. Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Knowledge-based systems recommender systems survey. **46**, 109–132 (2013)
10. Zhang, Q., Lu, J., Zhang, G.: Recommender systems in E-learning. J. Smart Environ. Green Comput. **1**(2), 76–89 (2021). https://doi.org/10.20517/jsegc.2020.06
11. Gulzar, Z., Leema, A.A., Deepak, G.: PCRS : personalized course recommender system based on hybrid approach. Procedia Comput. Sci. **125**, 518–524 (2018). https://doi.org/10.1016/j.procs.2017.12.067
12. Madani, Y., Erritali, M., Bengourram, J., Sailhan, F.: Social collaborative filtering approach for recommending courses in an e-learning platform. Procedia Comput. Sci. **151**, 1164–1169 (2019)
13. Morsomme, R., Alferez, S.V.: Content-based course recommender system for liberal arts education. In: International Educational Data Mining Society (2019)
14. Roy, D., Dutta, M.: A systematic review and research perspective on recommender systems. J. Big Data **9**, 59 (2022). https://doi.org/10.1186/s40537-022-00592-5
15. Bhattacharya, S., Sarkar, D., Kole, D.K., Jana, P.: Recent Trends in Recommendation Systems and Sentiment Analysis. Dans Elsevier eBooks, pp. 163–175 (2022). https://doi.org/10.1016/b978-0-32-385708-6.00016-3
16. Bobadilla, J. et al.: Collaborative filtering adapted to recommender systems of e-learning. Knowl.-Based Syst. **22** (2009)
17. Wang, D., Liang, Y., Xu, D., Feng, X., Guan, R.: A content-based recommender system for computer science publications. Knowl.-Based Syst. **157**, 1–9 (2018)
18. Afoudi, Y., Lazaar, M., Al Achhab, M.: Hybrid recommendation system combined content-based filtering and collaborative prediction using artificial neural network. Simul. Modelling Pract. Theory **113**, 102375 (2021)
19. Son, J., Kim, S.B.: Content-based filtering for recommender systems using multi-attribute networks. Expert Syst. Appl. **89**, 404–412 (2017). https://doi.org/10.1016/j.eswa.2017.08.008
20. Pazzani, M.J., Billsus, D.: Content-Based Recommendation Systems. Brusilovsky (2007)
21. Brusilovski, P., Kobsa, A., Nejdl, W. (eds): The Adaptive Web. Lecture Notes in Computer Science, vol 4321. Springer, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72079-9_10
22. Das, C., Sahoo, A.K., Pradhan, C.: Multicriteria recommender system using different approaches. In: Cognitive Data Science in Sustainable Computing, Cognitive Big Data Intelligence with a Metaheuristic Approach. Academic Press, Elsevier, pp. 259–277 (2022). https://doi.org/10.1016/B978-0-323-85117-6.00011-X

23. Sammut, C., Webb, G.I.: Collaborative Filtering. In: Encyclopedia of Machine Learning. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-30164-8_138
24. Isinkaye, F.O., Folajimi, Y.O., Ojokoh, B.A.: Recommendation systems: principles, methods and evaluation. Egypt. Inform. J. **16**, 261–273 (2015). https://doi.org/10.1016/j.eij.2015.06.005
25. Kluver, D., Ekstrand, M.D., Konstan, J.A. Rating-Based Collaborative Filtering: Algorithms and Evaluation. Social Information Access. Springer, pp. 344–390 (2018)
26. Khojamli, H., Razmara, J.: Survey of similarity functions on neighborhood-based collaborative filtering. Expert Syst. Appl. **185**, 115482 (2021). https://doi.org/10.1016/j.eswa.2021.115482
27. Masthoff, J., Mobasher, B., Desmarais, M.C., Nkambou, R., Guo, G.: Resolving data sparsity and cold start in recommender systems. In: User Modeling, Adaptation, and Personalization. UMAP 2012. Lecture Notes in Computer Science, vol 7379. Springer, Berlin, Heidelberg (2012). https://doi.org/10.1007/978-3-642-31454-4_36
28. Kotkov, D., Veijalainen, J., Wang, S.: How does serendipity affect diversity in recommender systems? A serendipity-oriented greedy algorithm. Computing **102**, 393–411 (2020). https://doi.org/10.1007/s00607-018-0687-5
29. Murat, G., Sule, G.O.: Combination of web page recommender systems. Exp. Syst. Appl. **37**(4), 2911–2922 (2010)
30. Mican, D., Tomai, N.: Association-rules-based recommender system for personalization in adaptive web-based applications. In: Daniel, F., Facca, F.M. (eds.) Current Trends in Web Engineering. ICWE 2010. Lecture Notes in Computer Science, vol. 6385. Springer, Berlin, Heidelberg (2010). https://doi.org/10.1007/978-3-642-16985-4_8
31. Aggarwal, C.C.: Knowledge-based recommender systems. In: Recommender Systems. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-29659-3_5
32. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook. Springer, Boston, MA (2015). https://doi.org/10.1007/978-1-4899-7637-6_6
33. Zhang, C.: Research on IMDB film score prediction based on improved whale algorithm. Procedia Comput. Sci. **208**, 361–366 (2022). https://doi.org/10.1016/j.procs.2022.10.051
34. TF–IDF. In: Sammut, C., Webb, G.I. (eds.) Encyclopedia of Machine Learning. Springer, Boston, MA (2011). https://doi.org/10.1007/978-0-387-30164-8_832
35. Li, B., Han, L.: Distance weighted cosine similarity measure for text classification. In: Yin, H., et al. (eds.) Intelligent Data Engineering and Automated Learning—IDEAL 2013. IDEAL 2013. Lecture Notes in Computer Science, vol. 8206. Springer, Berlin, Heidelberg (2013). https://doi.org/10.1007/978-3-642-41278-3_74
36. Kagita, V.R., Pujari, A.K., Padmanabhan, V., Sahu, S., Kumar, V.: Conformal recommender system. Inf. Sci. **405**, 157–174 (2017). https://doi.org/10.1016/j.ins.2017.04.005
37. Winoto, P., Tang, T.Y.: The role of user mood in movie recommendations. Expert Syst. Appl. **37**(8), 6086–6092 (2010). https://doi.org/10.1016/j.eswa.2010.02.117

Gamification as a Teaching Strategy for Enhancing Math Problem-Solving Skills in AI: A South African Perspective

Janine Olivier, Anass Bayaga, and Greyling Jean

Abstract This article explores the transformative potential of gamified coding instruction in enhancing learners' computational thinking and mathematical skills. Central to this investigation is the Tanks coding game, a product of Nelson Mandela University, acclaimed for introducing coding to over 100,000 learners in South Africa. The game's unique adaptability, requiring no reliance on computer labs, electricity, or internet connectivity, positions it as an innovative solution for addressing educational inequalities. Anecdotal evidence from educators underscores the positive impact of gamification on learners' mathematics experience and success. In anticipation of future research, this study outlines plans for researchers to implement Tanks and delving into the real-world application of gamified coding tools. The outcomes seek to contribute nuanced insights that extend beyond anecdotal evidence, informing educational practices and advancing the discourse on leveraging technology and gamification for inclusive and effective learning experiences in diverse settings.

Keywords Gamification. AI · Digitisation · Education · Higher-order thinking

1 Introduction

The advent of digitalization and artificial intelligence (AI) has ushered in a profound transformation in societal functioning, holding the potential for both developmental improvement and the exacerbation of societal inequalities within countries. Navigating this dynamic landscape requires a proactive approach to identify and forecast skills demands, educational opportunities, and responses within the educational value system. In South Africa, the teaching, learning, and assessment of 21st-century (21C) skills are guided by the National Curriculum Statement Grades R-12, which

J. Olivier · G. Jean
Nelson Mandela University, Gqeberha, South Africa
e-mail: Jean.Greyling@mandela.ac.za

A. Bayaga (✉)
University of the Western Cape, Gqeberha, South Africa
e-mail: abayaga@uwc.ac.za

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_23

underscores the government's commitment to a digitalization strategy [4, 8]. This commitment aims to prepare learners to be contributors in the 21C digital society and AI age, emphasizing inclusivity and the acquisition of knowledge and skills. Educators play a crucial role in addressing diversity and barriers to learning within the digital society and AI economy. Computational Thinking (CT), a digital literacy thought process and problem-solving approach, emerges as a key player in offering learners opportunities for knowledge and skill development. Researchers highlight its potential to promote programming skills, behavior, and support mathematical learning and higher-order thinking, all essential skills within the AI marketplace. The intrinsic mathematical nature of AI underscores the importance of mathematical thinking for effective understanding, functioning, and problem-solving in modern society.

Despite the significance of mathematics in technological, social, economic, and scientific development, concerns persist regarding a decline in maths achievement and engagement, particularly in the middle years of education [5]. AI's pivotal role in a country's economic development further amplifies the urgency for educational adaptation.

Research suggests that block-based programming, such as the game Tanks, is an effective way to introduce coding to learners, particularly those in underprivileged, remote, and marginalized communities [2]. Tanks, with its unplugged design, addresses infrastructure limitations and contributes to the development of skills like computational thinking and logic through a practical "learn by doing" approach. The humanistic approach underscores the significance of values and skills for effective human-machine collaboration in the globalized and complex AI environment [9]. Teaching methods supportive of the AI environment should foster the development of 21st-century skills, necessitating an urgent shift beyond rote learning and memorization particularly in developing countries where learning outcomes are a concern. Understanding the impact of integrating a coding game on learners' engagement and motivation in maths, as well as the transfer of skills from computational thinking to maths education, including the assimilation of a coding game on number pattern objectives through logico-maths knowledge, becomes imperative. Gamified strategies motivate 21C skills and AI development opportunities, providing learners with engaging challenges and rewards. AI emphasizes the transfer of problem-solving, mathematical, and reasoning skills to other disciplines essential for 21C skills. The perceived challenge of maths is deeply entrenched in learners' beliefs, influencing their perception of the subject and, consequently, their sense of capability. Addressing this challenge requires not only the integration of technological tools in maths instruction but also a focus on logical thinking and problem-solving skills [10]. A reciprocal relationship between motivation, engagement, and academic achievement further underscores the need for an engaging and interactive learning environment. Gamification emerges as a valuable strategy to enhance mathematical understanding, fostering engagement, motivation, social influence, and academic performance. This necessitates an evolution in teaching methods, information integration, and communication within the education system, moving beyond traditional content delivery toward innovative teaching strategies. Thus, the current study examines the question whether

previous research show that the integration of a coding game as a maths problem-solving teaching strategy could promote skills demands and teaching objectives of the AI age whilst mitigating the risk of aggravating societal inequalities? Response to such a question will assist in as a future research direction, propose the use of the block-based coding tool Tanks to enhance the mathematical and computational thinking skills of learners. This comprehensive approach aims to contribute valuable insights into the intersection of gamification, AI, and mathematics cognition.

1.1 Research Objective

In light of these considerations, there is a pressing need for research on CT-based mathematics instructions and the use of computing tools to support educator development. This includes exploring the integration of CT in mathematics instructions, understanding how CT supports mathematics learning, and vice versa.

- The proposed research seeks to determine whether the integration of a block-based coding game, such as Tanks can effectively address challenges in maths education by imparting essential computational thinking, critical thinking, and problem-solving skills.

2 Literature Review

In the contemporary information-based society, the demand for 21st-century knowledge, skills, and abilities is imperative to effectively navigate the challenges presented by the AI age. Knowledge itself is construed as mental representations, manifesting as concepts in various forms-be it social, physical, or logico-mathematical.

Specifically, logico-maths knowledge, representing relationships between objects, is acquired through reflective abstraction, wherein learners organize and interpret reality using physical and social knowledge as foundational elements [9, 11]. However, numerous challenges impede the development of mathematical understanding. Research underscores the influence of instructional approaches and tools on learners' perceptions, the psychological attitudes that shape interest in maths, and the presence of passive and distracted learners disengaged from the learning environment [7]. Conceptual maths knowledge, intricately linked to logical relationships between objects, tends to be underdeveloped, exacerbated by traditional teaching methods [1]. The challenges in maths learning are twofold, encompassing both instructional methods and learners' engagement levels. Active cognitive engagement is paramount for constructing and comprehending 21st-century concepts, necessitating a departure from traditional instructional approaches. Studies indicate a positive reception of gamification in addressing these challenges, attributing its success to fostering interest in maths learning through elements of contest, challenge, and entertainment [1]. The traditional approach, marked by a lack of understanding, is recognized as

a major impediment to conceptual maths knowledge development. Maths learning, crucial for understanding facts and acquiring problem-solving skills, requires a departure from rote learning towards real-world applications. Learners, inherently inclined to enjoy games, find comfort in game-based learning environments as they align with the desire to win and develop skills and strategies.

Areas for Future Research: While gamification shows promise in enhancing learners' attitudes and engagement in maths learning, there is a need for more nuanced research including but not limited to coding and computational thinking; mathematical thinking and mathematical and computational thinking. These are further explored shortly.

2.1 Gamification

Gaming has gained traction as a strategy to enhance engagement and overall experience in educational settings [1]. This involves the introduction of game-based learning and gamification, with the former focused on games explicitly designed to support learning objectives and the latter integrating game elements into educational subjects to achieve specific educational goals. Gamification transforms non-gaming entities into gaming entities, aiming to elicit constructive behavior and attitude changes, ultimately impacting motivation and engagement positively. Research indicates that gamified teaching strategies are designed to imbue the learning environment with meaning, fostering active engagement among learners. Through the application of rules and objectives, students are encouraged to collaboratively tackle challenges, communicate effectively, and develop a nuanced understanding of the context and learning outcomes. Learners actively participate in the learning process, applying knowledge through reflection, exploration, risk-taking, decision-making, and problem-solving within real-life situations. Games, as a part of this gamified approach, empower learners to take charge of their learning journey, promoting the development of problem-solving skills, resilience in the face of setbacks, and feedback-driven improvement. The facilitation of an atmosphere that encourages active and participatory learning, coupled with a gamification approach, provides a platform for learners to question, deliberate, reason, communicate, reflect, and critique. This not only enhances knowledge retention but also makes the learning experience exciting and appealing, fostering a non-threatening and engaging environment.

Learning experiences should guide learners to engage with ideas and explore relationships, with gamified learning proving beneficial for interdisciplinary skills such as critical thinking, interpersonal communication, collaboration, and debating.

Collaborative learning, facilitated by gamification, allows learners to gain knowledge through interactive information sharing in an enjoyable and symbiotic manner. Gamification's contextualized and personalized approach meets diverse learning needs, positively impacting learners' motivation to understand complex subjects and concepts. As learners explore various problem-solving approaches, creative

and critical thinking skills are honed. This process assists in contextualizing ideas, linking them with structures and other fields, ultimately fostering a positive attitude toward learning. With gamification, reluctance to learning diminishes, and learners' performance improves, emphasizing the significance of this approach in modern educational contexts.

- Research Gaps and Future Directions: While gamification shows promise in enhancing learner engagement and outcomes, there are gaps in the existing research that warrant further investigation:
- Effectiveness Across Educational Levels: Research should explore how gamification impacts learners at different educational levels, from primary to tertiary education, considering developmental and cognitive differences.
- Long-Term Impact on Learning: Investigating the long-term effects of gamification on knowledge retention and application as learners progress through their academic journeys.
- Teacher Training and Implementation: Examining the role of teacher training in effectively implementing gamification strategies and addressing challenges in diverse learning environments.
- Inclusive Design: Ensuring that gamified learning environments are designed inclusively, catering to learners with diverse learning needs and styles.
- Comparison with Traditional Methods: Conducting comparative studies to assess the effectiveness of gamification against traditional teaching methods and identifying the specific advantages and disadvantages of each approach.

Addressing these research gaps will contribute to a more nuanced understanding of the impact and potential limitations of gamification in education, guiding educators, policymakers, and researchers toward informed decision-making in the integration of gamified strategies into educational practices.

2.2 Coding Skills in the 21st Century: Bridging Gaps and Fostering Computational Thinking

In the twenty-first century, coding skills have become indispensable, primarily attributed to the pervasive influence of Information and Communication Technologies (ICT) in the AI age and the digital society [12]. Coding, or computer programming, involves the creation and design of software programs, employing specific programming languages to instruct computers in problem-solving tasks. The significance of coding lies in its ability to empower individuals to communicate effectively with computers and devise solutions tailored to contemporary challenges. Coding is typically taught through both plugged activities, involving computer implementation, and unplugged activities, which are conducted without the aid of computers. The unplugged approach is particularly valuable in environments lacking technological infrastructure, such as disadvantaged schools and rural areas, overcoming barriers to

teaching and learning programming and AI [12]. Unplugged coding activities play a pivotal role in imparting fundamental computational thinking concepts, fostering skill development, injecting an element of entertainment into lessons, and addressing potential misconceptions or negative attitudes toward programming [12]. Importantly, this approach proves less intimidating for teachers lacking programming backgrounds, thereby facilitating the enhancement of students' problem-solving skills in a cost-effective manner.

Computational Thinking (CT) emerges as a foundational skill set embedded in the coding paradigm. CT involves the thought process of articulating problems, formulating solutions, and presenting executable solutions for computers [15]. Coding, as a manifestation of CT, employs a problem-solving approach aligned with computer science principles [11]. In educational settings, learners are introduced to CT through coding, fostering the development of higher-order cognitive abilities crucial for both the twenty-first century and AI skills [11]. CT entails breaking down a problem into manageable parts, developing generic solutions through decomposition, identifying patterns and variables, and formulating algorithmic solutions—core principles reflected in the organization of code in loops and sequences involving defined variables [12]. Decomposition involves simplifying a problem into smaller parts, pattern recognition identifies and applies patterns to problem-solving steps, generalization uses identified equations for predictions, and abstraction determines patterns, generalizes objects, and identifies similarities among objects. The algorithm serves as the structural guide directing tasks to solve problems or achieve goals. In the realm of AI and 21st-century learning, problem-solving skills are paramount, and CT emerges as a conduit to acquiring analytical and critical thinking skills essential for interdisciplinary problem-solving [12]. Recognizing the interconnectedness of coding, CT, and broader skill development, the integration of coding skills into educational curricula becomes imperative for preparing learners to navigate the challenges of the evolving digital landscape.

Research Gaps and Future Directions: While the integration of coding and CT into education holds promise, there are research gaps that necessitate further exploration:

- Long-Term Impact of Unplugged Coding: Investigating the enduring effects of unplugged coding activities on learners' computational thinking and problem-solving abilities over an extended period.
- Comparative Analysis of Plugged vs. Unplugged Coding: Conducting comparative studies to evaluate the effectiveness of plugged and unplugged coding activities in diverse educational settings.
- Teacher Preparedness and Training: Exploring the role of teacher training in enhancing educators' preparedness to teach coding and computational thinking, addressing potential challenges.
- Inclusivity in Coding Education: Assessing the inclusivity of coding education, particularly in reaching learners from underprivileged backgrounds or regions lacking technological infrastructure.

- Impact of Coding on Interdisciplinary Skills: Investigating how the integration of coding skills and computational thinking influences the development of interdisciplinary skills such as critical thinking and collaboration.

By addressing these research gaps, educators, policymakers, and researchers can refine and optimize coding education strategies, ensuring they align with the evolving needs of learners in the twenty-first century.

2.3 Enhancing Mathematical Thinking Through Gamification: A Critical Imperative for 21st Century Learning

Mathematical Thinking (MT) stands as a cognitive process intricately woven into operations, processes, and dynamics, deploying mathematical skills to unravel problems marked by contradiction, anxiety, and surprise. As an indispensable skillset for navigating the complexities of AI and the information-based society, the application of mathematics in problem-solving underscores the critical need to enhance mathematical thinking [12]. In the context of an ever-evolving educational landscape, the gamified environment emerges as a fertile ground for honing mathematical skills and potentially transforming attitudes toward mathematics. By infusing physical materials and interactive elements, gamification provides a tangible platform wherein learners can naturally relate to and comprehend the intricacies of mathematical tasks. This approach proves pragmatic in mathematics teaching, offering a spectrum of cognitive and affective learning methods that cater to diverse learning styles. The gamification of mathematics instruction becomes a strategic endeavor, not merely for the transmission of mathematical knowledge but for crafting positive and engaging learning experiences. It is imperative that mathematics teaching transcends traditional boundaries and becomes contextually relevant. A positive encounter with mathematics, facilitated through gamification, has the potential to reshape learners' attitudes toward the subject, making it more accessible and enjoyable. Learners' conceptions, attitudes, and expectations regarding mathematics form the bedrock of their learning experiences and achievements. Mathematics educators, recognizing the pivotal role they play, can leverage gamification as a powerful tool to actively engage learners, fostering an understanding of critical math skills that goes beyond rote memorization [12]. This engagement not only bolsters mathematical thinking but also has the capacity to enhance visual learning, making mathematical concepts more tangible and comprehensible.

Research Gaps and Future Directions: While the integration of gamification in mathematics education holds promise, there exist research gaps that warrant further exploration:

- Long-Term Impact of Gamified Mathematics: Investigating the enduring effects of gamified mathematics instruction on learners' mathematical thinking and attitudes toward the subject over an extended period.
- Comparative Analysis of Gamification Strategies: Conducting comparative studies to assess the effectiveness of different gamification strategies in diverse educational settings.
- Influence of Gamification on Visual Learning: Exploring the specific ways in which gamification influences visual learning in the context of mathematical thinking.
- Teacher Training in Gamified Mathematics: Assessing the impact of teacher training programs on effectively integrating gamification into mathematics instruction.
- Inclusive Gamification: Examining the inclusivity of gamified mathematics instruction in reaching learners with diverse learning styles, abilities, and backgrounds.

By delving into these research gaps, educators and researchers can further refine gamified mathematics instruction, ensuring its alignment with the diverse needs and challenges posed by the 21st-century learning landscape.

2.4 Synergies in Computational Thinking (CT) and Mathematical Thinking (MT): A Pedagogical Nexus

Computational Thinking (CT) and Mathematical Thinking (MT) emerge as intricately intertwined and mutually reinforcing problem-solving methodologies, sharing a tapestry of similarities. MT, primarily employed to resolve mathematical problems through questioning, absorption, and review, finds its counterpart in CT, denoting a problem-solving approach executable by a computer. The bedrock of CT includes decomposition, abstraction, algorithmic design, debugging, and pattern recognition, processes that resonate with the iterative design, correction, and review inherent in MT [14]. Both CT and MT are symbiotic, fostering mutual growth through practice, reflection, and the cultivation of aspects fundamental to creative thinking. MT finds application within computational tools, while CT operates within the domain of mathematics. The reciprocal exchange of information between CT and MT not only supports learners in recognizing the interconnections between the two fields but also amplifies their problem-solving capabilities [14]. In the realm of problem-solving, CT empowers learners to deploy diverse methodologies in tackling cognitive challenges. This encompasses formulating problems conducive to computer and tool utilization, analyzing and organizing data, representing data through abstractions, automating solutions via algorithmic thinking, and identifying, analyzing, and implementing solutions. The confluence of coding and mathematics, rooted in shared conceptual foundations, heralds an integrated pedagogical approach. Logical thinking, pivotal in both CT and mathematics education, becomes a linchpin for the development of AI

skills, fostering critical attributes such as communication, critical thinking, problem-solving, autonomy, analysis, and creativity essential for AI [14]. The integration of mathematics and computational thinking underscores the significance of computation and digital technologies across disciplines. The educational outcomes of both MT and CT converge, advocating the utilization of technological tools to deepen conceptual understanding. The coding process becomes a catalyst influencing the development of mathematical thinking [3], intertwining with the understanding of mathematical ideas essential for AI skills. In their capacity as problem-solving methodologies, CT and MT exhibit shared behaviors, including abstract thinking, metacognition, trial and error, ambiguity tolerance, flexibility, and the ability to contemplate diverse problem-solving methods. Computational tools and skills breathe new life into mathematics learning, offering novel techniques for solving complex problems. Reciprocally, mathematics provides a fertile context for CT, with both concepts serving as problem-solving approaches crucial for comprehending the symbolic and systematic representation processes integral to AI contexts [14]. Peer collaboration emerges as a cornerstone for task implementation, with communication and social interaction acting as facilitators for CT and MT learning.

3 Discussion

Based on the literature thus far, facilitating learning through gamification is crucial for several reasons and from theoretical stance. Gamification, as a teaching strategy, serves as a catalyst for transformative learning experiences, actively engaging learners in the Tanks coding game and contributing to their knowledge construction and skills development [6]. Within the constructivist framework, where knowledge is seen as actively created, gamified environments act as fertile grounds for learners to access new structures, interpretations, and manipulations of knowledge [6]. In a gamified classroom, the teacher assumes the role of a facilitator, reshaping the traditional approach to one of active involvement and contextual material connections. This shift supports the construction of physical and logico-mathematical knowledge, unattainable through passive learning. The facilitator's role encourages negotiation and active participation, a departure from the conventional 'passive and inactive' mode of learning. Constructivist theories align with this view, emphasizing interactive opportunities within gamified environments that nurture reflection, creativity, and reasoning through problem-solving processes [6]. Diverging from traditional teaching methods, the gamified approach in mathematics instruction taps into curiosity, aligning the subject with learners' interests and activating responses based on sensory signals. This modification of knowledge, assimilating visual and tactile data, leads to accommodation as learners build upon prior knowledge for new learning. Constructivism, intrinsic to the educational context and active learning theory, champions the creation of knowledge through active participation and engagement [13]. The social dimension of learning is elevated within the

constructivist classroom, where learners discover new ideas and construct knowledge through social interaction. Collaborative learning unfolds through engagement in block-based coding gamification, accommodating diverse learners and shifting learning towards participation in social practice. The contextual and experiential nature of learning within this framework underscores the importance of problem-solving, motivation, collaboration, and making connections, all integral to academic success and enhanced mathematical learning. Abstraction, logical thinking, innovation, and creativity central to computational thinking resonate with the objectives of AI.

The gamified coding approach in maths problem-solving allows learners to make connections, integrating skills from various disciplines. The application of CT in a gamified manner guides learners to apply knowledge gained to define and solve problems, demonstrating the potential for interdisciplinary learning.

4 Conclusion

In summary, the amalgamation of coding instruction with gamification emerges as a potent tool for augmenting learners' computational thinking and mathematical proficiency. The Tanks coding game, a product of Nelson Mandela University in South Africa, has notably introduced coding to over 100,000 learners across the nation, offering an innovative solution tailored to the African context. Particularly noteworthy is its adaptability, requiring no reliance on computer labs, electricity, or internet connectivity when deployed in workshops. The transformative potential of gamification, as evidenced by the Tanks coding game, is underscored by anecdotal evidence from educators, indicating a positive impact on learners' engagement and success in mathematics. This approach not only enhances skills but also plays a crucial role in addressing the existing disparities within the AI age, contributing to the preparation of learners for the challenges and opportunities that lie ahead.

5 Future Research

This future research could aim to provide nuanced insights into the practical implications, challenges, and successes associated with integrating gamified coding tools in the unique educational landscape of rural schools. By doing so, researchers may aspire to contribute valuable knowledge that extends beyond anecdotal evidence, fostering a comprehensive understanding of the sustained impact and potential optimizations of the Tanks coding game in enhancing computational thinking and mathematical skills. The outcomes such study will not only inform educational practices but also contribute to the ongoing discourse on leveraging technology and gamification for inclusive and effective learning experiences in diverse settings.

References

1. Adipat, S., Laksana, K., Busayanon, K., Asawasowan, A., Adipat, B.: Engaging students in the learning process with game-based learning: the fundamental concepts. Int. J. Technol. Educ. **4**(3), 542–552 (2021). https://doi.org/10.46328/ijte.169
2. Batteson, B.: Investigation and Development of an Inexpensive Educational Tool Suite for an Introduction to Programming. Honours Treatise, Nelson Mandela University (2017)
3. Benton, L., Kalas, I., Saunders, P., Hoyles, C., Noss, R.: Beyond jam sandwiches and cups of tea: an exploration of primary pupils algorithm-evaluation strategies. J. Comput. Assist. Learn. **34**, 590–601 (2018). https://doi.org/10.1111/jcal.12266
4. Department of Basic Education (DBE). The National Curriculum Statement Grades R-12. Pretoria: Department of Basic Education (2011)
5. Durksen, T., Way, J., Bobis, J., Anderson, J., Skilling, K., Martin, A.: Motivation and engagement in mathematics: a qualitative framework for teacher-student interactions. Math. Educ. Res. J. **29**(2) (2017). https://doi.org/10.1007/s13394-017-0199-1
6. Greyling, J., Tokosi, T.: Investigating the roles and competence of a Tangible game facilitator. Paper presented at SAICSIT'20: Conference of the South African Institute of Computer Scientist and Information Technologies. Cape Town, South Africa (2020). https://doi.org/10.1145/3410886.3410894
7. Hagan, J.E., Amoaddai, S., Lawer, V.T., Atteh, E.: Students' perception towards mathematics and its effects on academic performance. Asian J. Educ. Soc. Stud. **8**(1), 8–14 (2020). https://doi.org/10.9734/AJESS/2020/v8i130210
8. Hamlen, K.R.: Understanding children's choices and cognition in video game play a synthesis of three studies. Z. Fur Pscychol. **221**(3), 107–114 (2013). https://doi.org/10.1027/2151-2604/a000136
9. Kim, S., Raza, M., Seidman, E.: Improving 21st-century teaching skills: the key to effective 21st-century learners. Comp. Int. Educ. **14**(1), 99–117 (2019). https://doi.org/10.4018/jig.2011070103
10. Mthethwa, M., Bayaga, A., Bosse, M., Williams, D.: GeoGebra for learning and teaching: a parallel investigation. S. Afr. J. Educ. **40**(2), 1–12 (2020). https://doi.org/10.157/saje.v40nza1669
11. Roman-Gonzalez, M., Perez-Gonzalez, J., Jimenez-Fernandez, C.: Which cognitive abilities underlie computational thinking? Criterion validity of the computational thinking test. Comput. Hum. Behav. **72**(2), 678–691 (2017). https://doi.org/10.1013/j.chb.2016.08.047
12. Tsarava, K., Moeller, K., Pinkwart, N., Butz, M., Trautwein, U., Ninaus, M.: Training computational thinking: game-based unplugged and plugged-in activities in primary school [Conference Presentation]. In: 11th European Conference in Game-Based Learning (2017)
13. Vygotski, L.: Mind in Society. Development of Higher Psychological Processes. Harvard University Press (1978)
14. Wu, W., Yang, K.: The relationships between computational and mathematical thinking: a review study on tasks. Cogent Educ. **9**(1), 1–19 (2022). https://doi.org/10.1080/23386X.2022.2098929
15. Ye, H., Liang, B., Ng, O., Chai, C.: Integration of computational thinking in K-12 mathematics education: a systematic review on CT-based mathematics instruction and student learning. Int. J. STEM Educ. **10**(3), 1–26 (2023). https://doi.org/10.1186/s40594-023-00396-w

Exploring the Impact of Artificial Intelligence in Education: A Comprehensive Review and Future Directions

Said Ouabou and Abdellah Idrissi

Abstract Artificial intelligence (AI) is revolutionizing education by offering new possibilities to enhance learning and teaching practices. This paper reflects on the importance of AI in education and explores its implications for the future. AI enables personalized learning experiences, expands access to education, improves teacher efficiency, supports lifelong learning, and predicts educational trends. However, ethical concerns and the irreplaceable role of teachers must be carefully considered. The paper outline the need for future research and continuous exploration of AI in education to fully understand and harness its potential, while ensuring ethical and beneficial use for all learners.

Keywords Artificial intelligence · Education · Personalized learning · Access to education

1 Introduction

Artificial intelligence (AI) in education is marked by a convergence between technological advances and the growing needs of the education field [1]. The main goal of integrating AI into education is to improve the learning experience of learners by providing resources and activities tailored to their individual needs [2]. Overall, the context of AI in education is marked by increasing research, experimentation and pilot deployments aimed at harnessing the potential of AI to improve learning and teaching processes [3].

The objective of this paper is to examine the impact of artificial intelligence in the field of education. Studying the impact of AI in education is of paramount

S. Ouabou (✉) · A. Idrissi
Artificial Intelligence & Data Science Group, Faculty of Science, IPSS Team, Mohammed V University in Rabat, Rabat, Morocco
e-mail: said-ouabou@um5r.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_24

importance in several respects. Understanding how AI can be optimally used in education is essential to harnessing its potential and addressing the challenges that arise. Furthermore, the present study contributes to the existing literature on AI in education by providing a comprehensive and up-to-date synthesis of knowledge. This in-depth review will identify current trends, gaps, and challenges in the use of AI in education and pave the way for new research and innovative applications.

This paper aims to explore the impact of AI in education, highlighting its potential benefits, challenges, and prospects.

2 Theoretical Foundations of AI in Education

2.1 Definition of Artificial Intelligence

Artificial intelligence (AI) is a field of technological research and development that aims to create computer systems capable of performing tasks that normally require human intelligence [4].

The goal of AI is to replicate some of the cognitive abilities, such as perception, reasoning, learning, problem solving, natural language understanding, and decision making [5].

AI systems are designed to be used in collaboration with humans, providing support, automating certain tasks and offering analysis and recommendations to make informed decisions [1].

2.2 AI Applications in Education

Artificial intelligence has many potential applications in education. They can analyze learner performance, identify their gaps, and provide recommendations and educational resources tailored to their specific needs. These systems can also offer adaptive assessments, adjusting question difficulty based on learner responses. They use machine learning algorithms to evaluate learner responses, providing immediate and accurate feedback [4, 6].

This helps teachers save time and learners get faster assessments. These analytics enable teachers and educational leaders to gain in-depth insights into the learning process, identify trends, issues, and individual needs, and make informed decisions to improve teaching and learning [7].

2.3 *Potential Benefits of AI in Terms of Learning and Teaching*

Artificial intelligence offers several potential benefits in terms of learning and teaching.

Personalization of learning: AI enables the personalization of learning by adapting educational content, methods and resources to the individual needs and preferences of each learner. This promotes more effective learning by avoiding redundancies and providing appropriate challenges at each skill level [8].

Distance access and e-learning: AI can facilitate access to education by enabling distance and online learning. AI systems can be used to develop interactive and engaging online learning platforms, providing learners in disadvantaged communities with access to quality learning resources, regardless of their geographical location or logistical constraints.

Translation and language assistance: in cases where the learners' mother tongue is distinct from the language used in the educational context, artificial intelligence has the potential to enable machine translation and provide linguistic support. AI systems enable the design of real-time translation aids, automated subtitles or virtual tutoring initiatives, enabling learners from marginalized backgrounds to understand and actively participate in educational activities.

Data Analysis Process to Identify Needs: Artificial Intelligence (AI) can be used to examine large amounts of learner data and determine the distinct needs of learners from disadvantaged communities. By using machine learning techniques, AI can recognize patterns and trends that can help educators understand the barriers and challenges faced by these learners; allowing them to adjust their teaching methods to meet their needs.

AI has the potential to be used in the creation of virtual tutoring systems and educational chatbots. By responding to inquiries, providing additional clarification, and helping learners in disadvantaged communities, these tools can provide additional support. Therefore, autonomous learning can be strengthened and continued access to appropriate educational support can be guaranteed.

Immediate feedback: AI systems can provide immediate feedback to learners, whether to correct their answers, guide them in solving problems, or provide feedback on their performance. This broadens access to knowledge and information, and promotes deeper exploration of topics of interest [9].

Teacher support: AI can support teachers by automating certain administrative tasks, such as correcting homework, managing grades, and generating reports. This allows them to spend more time on essential educational activities, such as interacting with learners, designing lesson plans, and personalizing instruction [10] (Fig. 1).

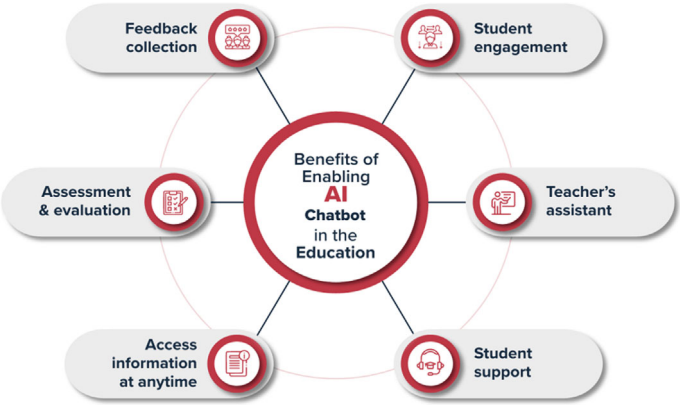


Fig. 1 Benefits of enabling AI chatbot in education [11]

2.4 Issues and Concerns Related to the Use of AI in Education

The use of artificial intelligence (AI) in education also raises several issues and concerns. First, it is crucial to ensure the privacy and security of learner data, ensuring that it is not misused or disclosed to unauthorized third parties. For example, automated assessment systems may be less accurate in assessing the responses of learners from minority groups, which can lead to inequities in learning. Transparency and understandability of AI systems are also major concerns. It is important to make the processes and results of AI systems transparent and explainable, so that teachers, learners and educational leaders can understand how decisions are made and have confidence in the results obtained.

AI systems provide learners with different types of feedback. Through the analysis of grammar, spelling and syntax of answers, these systems allow the automatic correction of assignments, providing immediate and accurate feedback that promotes autonomous learning. In addition, AI offers adaptive virtual tutoring, which consists of identifying the strengths and weaknesses of each learner and providing personalized assistance throughout their academic progress.

Additionally, the IA is used to assess students' proficiency in a variety of subjects, including mathematics, by offering in-depth feedback to help with skill improvement and the understanding of reasoning errors. Systems of artificial intelligence also analyze learners' writing, pointing out grammatical errors and suggesting revisions to enhance writing skills. Finally, the IA uses computer vision and natural language processing techniques to provide multimodal comments, such as in the field of visual arts, providing a detailed feedback on the creative work of the learners. Overall, these examples show how personalized, immediate, and detailed feedback is provided by AI systems to learners to help them recognize their mistakes, build their skills, and advance.

Table 1 Examples of AI implementation in educational contexts

Application of AI	Examples
1. Intelligent tutoring systems The intelligent	Tutoring system "Cognitive Tutor" developed by Carnegie Learning
2. Educational content recommendation systems	The recommendation system used by the Khan Academy online learning platform
3. Automated assessment	EdX uses AI for automated assessment of learner responses on its online learning platform
4. Educational chatbots	"Duolingo", which uses a chatbot to teach foreign languages
5. Speech and handwriting recognition	Companies such as Nuance Communications and Google use this technology in their educational products
6. Learning data analytics	Online learning platforms like Coursera use data analytics to evaluate learner performance and improve their courses

Finally, it is essential to provide adequate support and training to teachers to help them understand, use and critically evaluate AI systems.

3 Case Studies and Research Findings

3.1 Concrete Examples of AI Implementation in Educational Contexts

See Table 1.

3.2 Impacts Observed on Learners and Teachers

See Table 2.

3.3 Performance Evaluation of Educational AI Systems

To measure the effectiveness and ensure the appropriate use of AI in education, several key aspects must be taken into account [9]. First, instructional effectiveness should be assessed by analyzing the learning outcomes of learners using these systems compared to those using traditional teaching methods. Next, the adaptability and

Table 2 Impacts observed on learners and teachers

Impacts observed on learners:	Impacts observed on teachers:
1. Improved motivation and engagement: Adapting content and activities to the individual needs of learners, as well as instant feedback, promotes a more engaging and rewarding learning experience	1. Improved feedback and monitoring: AI systems provide insights into individual performance, common errors, and areas of reinforcement, making it easier to provide personalized feedback and monitor learner progress
2. Acquisition of transversal skills: AI-powered learning environments encourage learners to actively engage in interactive activities and make informed decisions, which promotes the development of these key skills	2. Professional development: By exploring and using AI-based tools, teachers can learn new technological and teaching skills, which helps them stay up to date with the latest advancements in teaching and improve their teaching practice

personalization of educational AI systems should be examined to determine their ability to adapt to the individual needs of learners and provide a personalized learning experience. The reliability and accuracy of educational AI systems are also important evaluation criteria. The quality of the recommendations, assessments and feedback provided by these systems should be verified by performing manual and independent assessments to ensure the accuracy of the automatically assessed responses, the relevance of the recommendations and the consistency of the feedback provided [12]. Furthermore, the transparency and explainability of the decision-making processes of educational AI systems must be assessed. It is essential to understand how these systems make their decisions, what algorithms are used and how recommendations or ratings are generated. It is important to detect possible biases in recommendations and assessments, protect learner privacy, ensure data security, and establish accountability for misuse of educational AI systems.

4 Future Challenges and Opportunities

4.1 Ethical Issues and Responsibility in the Use of AI in Education

The use of artificial intelligence (AI) in education raises ethical questions and highlights the need to define clear responsibilities. Among ethical issues, algorithmic bias must be taken into account in order to ensure equity in education and avoid reinforcing existing prejudices [7]. Learner privacy and data security are also major concerns, requiring adequate safeguards and compliance with data protection regulations. Transparency and explainability of educational AI systems are essential to enable users to understand the decisions made and trust the results and suggestions.

Responsibility for AI-based decisions needs to be clarified, emphasizing the importance of human oversight and ethical decision-making. Equity and access for all learners must be guaranteed, so that AI does not widen existing inequalities. Finally, training teachers is very important to enable them to understand and use AI ethically, as well as to support learners in their interaction with these systems [13].

More specific examples of potential ethical concerns related to the use of AI in education include:

Data protection: The use of AI in education generally requires the collection and analysis of large amounts of student data. Protecting the privacy of students and securing their personal data are major ethical issues. It is essential to ensure that sensitive information is not misused or disclosed to unauthorized third parties.
Algorithmic bias: Artificial intelligence systems use algorithms to make decisions and provide recommendations. However, these algorithms can be biased, resulting in the reproduction of existing biases in the training data. If the data used to form the system shows an imbalance in terms of gender, race or social class, AI may reproduce these inequalities and perpetuate discrimination.
Excessive automation: By using artificial intelligence systems to automate administrative tasks, teachers can save time, but it can also have a negative impact on the educational experience. Too much automation can interfere with students' social interactions, engagement and attention, which can negatively affect their learning and well-being.
Unequal access: While AI has the potential to improve access to education, there is a risk of reinforcing existing inequalities. Disadvantaged communities may be excluded from AI technologies due to the high cost of infrastructure and resources required. In addition, learners from disadvantaged social backgrounds may not benefit from the benefits of artificial intelligence if programs are not tailored to their specific needs.
Transparency and accountability: Decisions are made using complex and opaque models in artificial intelligence systems. This raises questions about transparency and accountability. It is essential that teachers, students and parents understand the decisions made by artificial intelligence systems, so that they can question and challenge them if necessary. It is also important to determine who is responsible for errors or damage caused by AI systems.

4.2 Teacher Training and Integration of AI into Educational Practices

To fully realize the benefits of AI in education. Educators must acquire the skills necessary to understand and operate educational AI systems effectively and ethically. Teacher training should include a thorough understanding of AI principles, key concepts such as machine learning and natural language processing, as well as practical knowledge on how to integrate AI into teaching activities. teaching

and learning. Teachers must also be able to analyze data generated by AI systems, understand their limitations, and interpret the results critically [5, 14].

Educators can benefit from continuing education specifically focused on integrating AI into education. This may include workshops, seminars or professional development programs to help them understand the key concepts of AI, Explore best practices and acquire technical skills to use AI tools and technologies.

They can also access specialized educational resources that provide concrete examples of the use of AI in different disciplines and educational contexts. These resources can include case studies, how-to guides, online tutorials and examples of AI-based learning projects. They can help educators understand how to adapt and integrate AI into their own teaching practices.

Collaboration between teachers and AI experts can be beneficial to develop suitable training programs and provide ongoing support to teachers in their use of AI [15].

Professional networks and learning communities focused on AI in education allow educators to share experiences, learn from each other, discuss challenges and solutions, and stay up to date on the latest developments and best practices in integrating AI into education.

Integrating AI into educational practices offers many possibilities, such as personalizing learning, adapting to individual learner needs, providing instant feedback, and identifying gaps in understanding. However, it is important to emphasize that AI should not replace the role of teachers, but rather support them in their work by offering new resources and perspectives. Therefore, teacher training and continued engagement in professional development are essential for successful integration of AI into educational practices, ensuring that teachers can fully exploit the potential of AI to improve education. student learning.

4.3 Perspectives and Emerging Trends in the Use of AI in Education

The prospects for using artificial intelligence (AI) in education are exciting, with many emerging trends promising to transform the educational landscape.

Using data analytics and machine learning, AI systems can tailor content, activities, and assessments based on learners' individual needs, fostering a more personalized and engaging learning experience [3]. Additionally, AI facilitates the automation of certain administrative and repetitive tasks, allowing teachers to focus more on higher value-added activities, such as creating innovative educational resources and interacting with learners. AI can also encourage collaboration and social learning by facilitating connection between learners and supporting collaborative learning environments [16]. For example, chatbots and virtual assistants can help learners ask questions, solve problems, and get feedback in real time. Additionally, AI can help address educational inequality by providing increased access to quality education.

AI-based technologies can be used to provide educational resources to learners in remote or underserved areas, as well as to support the learning of learners with special needs. However, it is important to remain aware of the challenges and ethical questions surrounding the use of AI in education. It is essential to ensure transparency, fairness, privacy and accountability in the design and implementation of educational AI systems.

5 Conclusion

In conclusion, this paper offers valuable insights into the transformative potential of artificial intelligence (AI) in education. This comprehensive review highlights the different ways AI can improve teaching and learning, including providing personalized learning experiences, automating administrative tasks, fostering collaboration, and improving access to education. The paper also highlights the need to consider ethical considerations and implement AI responsibly in educational contexts, ensuring transparency, fairness, privacy and accountability. Additionally, it emphasizes the importance of equipping educators with the skills and knowledge to effectively integrate AI into their practices. Future directions presented in the paper emphasize continued collaboration among educators, researchers, and technology developers to maximize the benefits of AI while mitigating the risks. By harnessing the power of AI and combining it with educational expertise, we have the potential to create an inclusive and adaptive educational environment that empowers learners and prepares them for the challenges of the digital age. The future is very promising with this context of artificial intelligence applied to education as well as to any other field, particularly in [17–32]. In this context, several ideas can be adapted and improved.

References

1. Hwang, G.-J., Xie, H., Wah, B.W., Gašević, D.: Vision, challenges, roles and research issues of artificial intelligence in education. Comput. Educ. Artif. Intell. **1**, 100001 (2020). https://doi.org/10.1016/j.caeai.2020.100001
2. Huang, J., Saleh, S., Liu, Y.: A review on artificial intelligence in education. Acad. J. Interdiscip. Stud. **10**(3), 206 (2021). https://doi.org/10.36941/ajis-2021-0077
3. Limna, P., Jakwatanatham, S., Siripipattanakul, S., Kaewpuang, P., Sriboonruang, P.: A review of artificial intelligence (AI) in education during the digital era, **3** (2022)
4. Ouabou, S., Idrissi, A., Daoudi, A., Bekri, M.A.: School dropout prediction using machine learning algorithms. In: Idrissi, A., (ed.) Modern Artificial Intelligence and Data Science: Tools, Techniques and Systems. In: Studies in Computational Intelligence, pp. 147–157. Cham: Springer Nature Switzerland (2023). https://doi.org/10.1007/978-3-031-33309-5_12
5. Bearman, M., Ryan, J., Ajjawi, R.: Discourses of artificial intelligence in higher education: a critical literature review. High. Educ. **86**(2), 369–385 (2023). https://doi.org/10.1007/s10734-022-00937-2
6. Fengchun, M., Wayne, H., Huang, R., Zhang, H.: UNESCO, AI and education: A guidance for policymakers. UNESCO Publishing (2021)

7. Limna, P.: Artificial intelligence (AI) in the hospitality industry: a review article
8. Mitchell, M.: Why AI is harder than we think (2021). arXiv https://doi.org/10.48550/arXiv.2104.12871
9. Bozkurt, A., Karadeniz, A., Baneres, D., Guerrero-Roldán, A.E., Rodríguez, M.E.: Artificial intelligence and reflections from educational landscape: a review of ai studies in half a century. Sustainability **13**(2), Art. no. 2 (2021). https://doi.org/10.3390/su13020800
10. Shabbir, J., Anwer, T.: Artificial intelligence and its role in near future (2018). arXiv https://doi.org/10.48550/arXiv.1804.01396
11. Voutik, L.: Benefits of AI Chatbot in the Education: The Future of EdTech. Quytech Blog. Accessed 17 Jan 2024 [Online]. Available: https://www.quytech.com/blog/benefits-of-ai-chatbot-in-education-future-of-edtech/
12. Cui, W., Xue, Z., Thai, K.-P.: Performance comparison of an AI-based adaptive learning system in China. In: 2018 Chinese Automation Congress (CAC), pp. 3170–3175 (2018). https://doi.org/10.1109/CAC.2018.8623327
13. Zhai, X., et al.: A review of artificial intelligence (AI) in education from 2010 to 2020. Complexity **2021**, 1–18 (2021). https://doi.org/10.1155/2021/8812542
14. Su, J., Ng, D.T.K., Chu, S.K.W.: Artificial intelligence (AI) literacy in early childhood education: the challenges and opportunities. Comput. Educ. Artif. Intell. **4**, 100124 (2023). https://doi.org/10.1016/j.caeai.2023.100124
15. Artificial intelligence in education: the three paradigms—ScienceDirect. https://www.sciencedirect.com/science/article/pii/S2666920X2100014X. Accessed 02 Nov 2023 [Online]
16. Evolution and revolution in artificial intelligence in education. Int. J. Artif. Intell. Educ. https://doi.org/10.1007/s40593-016-0110-3. Accessed 02 Nov 2023 [Online]
17. Elhandri, K., Idrissi, A.: Parallelization of Top_k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
18. Elhandri, K., Idrissi, A.: Comparative study of Top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
19. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
20. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
21. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv preprint arXiv:1307.5910
22. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. Int. Conf. Big Data Adv. Wirel. Technol. (2016)
23. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF, 107–116 (2006)
24. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
25. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
26. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
27. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
28. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
29. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)

30. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
31. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless technologies (2016)
32. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. International Journal of Business Intelligence and Data Mining (2017)

A Multi-agent Approach for Intelligent and Cooperative Learning Systems

Khaireddine Bacha

Abstract Information and Communication Technologies (ICT), the Internet in particular, have in recent years invaded our daily lives, both personal and professional. After having interfered in many areas such as traditional commerce and administrations, the Internet is about to become the keystone of a new form of education. We proposed a prototype of computer-assisted language learning (CALL): case of the Arabic language using artificial intelligence techniques, automatic Arabic language processing techniques. It provides answers to questions considered fundamental for learning written Arabic. Thus, we have reaffirmed the important intersections between the research dynamics in CALL and research in the field of NLP.

Keywords TELA · NLP · CALL · Pedagogical activities · Arabic language

1 Introduction

The role of computing in language teaching has changed significantly in recent years. Technological and educational developments now allow us to better integrate computer technology into the learning process. Computer Assisted Language Learning (CALL) has contributed, and can still contribute, to the success of independent, motivating learning in authentic environments.

Future developments in networked communications, multimedia, and artificial intelligence will come together to reinforce this trend. It should be emphasized that, although it is essential to think about how CALL can be successfully integrated into language teaching, it must be taken into consideration that the machine can never take the place of the man [1].

With the computer progress, the field of the automatic processing of natural language (NLP) has experienced a large growth for most of the languages. For the Arabic, the tools from the domain of NLP, have been slow to integrate in the platforms of learn-ing languages. In effect, the field of this study, is very broad, there are

K. Bacha (✉)
Laboratory LaTICE, University of Tunis, Tunis, Tunisia
e-mail: khairi.bacha@gmail.com

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_25

more and more research and technologies that are concerned about the specificities of this language [2–9] and which offer tools necessary for the development of its automatic processing. We then propose our environment of the Arabic TELA. The proposed approach is based on the integration of tools NLP.

2 State of the Art of Applications and Learning Environments

Computer Assisted Language Learning (CALL) occupies a leading position in language teaching. The development of information and communication technologies for teaching. Certain techniques and methods are considered in language learning by researchers in order to diagnose states of knowledge and implement automatic natural language processing tools. Their work aims to build interactive environments to support learning.

Computer environments oriented towards language acquisition cover two distinct, but complementary, areas of research and development: First, environments allowing fundamental research on language acquisition through a interaction with the computer [10]. Then finalized learning environments for the acquisition of linguistic skills, in writing, reading, speaking, designed using multiple educational approaches [11].

2.1 CALL Systems

The computer, too, can provide assistance and in recent years we have seen a proliferation of Computer-Assisted Learning projects: The majority of research and systems in the field of computer-assisted language learning computer were developed for English, followed by Japanese, French and German [12]. Platforms have all shown the advantage of having used tools from NLP to remedy the shortcomings of traditional platforms [6, 8, 9]. Among those aimed at language students are:

- The PLATO system (Programmed Logic for Automated Teaching Operations), developed at the University of Illinois (United States) for French, Spanish, German, Russian, Hebrew, Latin and even Esperanto [1].
- KANDA in Tokyo, for English, German, French, Spanish and Chinese [2].
- The OPE project, at the University of Paris VII, for English [9].
- MIRTO, developed within the LIDILEM laboratory in Grenoble, allows the creation of educational activities, almost instantly, by exploiting the possibilities of NLP procedures and tools. MIRTO can generate activities intended for learners for learning several languages [1]. These achievements are of certain educational interest, but limited by the fact that they only apply to written language.

Certain techniques and methods are considered in language learning by researchers, in order to diagnose states of knowledge and implement tools for automatic natural language processing. Their work aims to build interactive environments to support learning.

These systems generally associate the activities they offer with adequate feedback which allows almost automatic correction of learners' responses. The activities that can be designed and carried out with CALL systems are varied. Mangenot has drawn up the following list of exercise categories with appropriate feedback [13].

Oral exercises for transforming or recording statements give learners the opportunity to record and listen to themselves, in order to acquire oral skills. Help indices are provided for self-correction.

2.2 Challenges of the CALL

Although CALL brings benefits to language teaching and learning, there are still some issues to be resolved [14]. The obstacles preventing complete integration of CALL in language learning can be classified into the following categories:

- Financial obstacles are the most frequently encountered obstacles during the deployment of the CALL. They include the cost of hardware, software, maintenance, and staff training. As for colleges and universities, if the cost of adopting CALL proves excessive, it will be considered a luxury item and will undoubtedly delay its use.
- The tide of change has arrived so quickly that it has destroyed what was considered the norm in the past and created new opportunities on the other hand. But, there is a natural tendency for anything traditional to resist change, not only due to the lack of theories and knowledge among some ancient teachers, but also the deeply ingrained traditional methods of teaching. Teachers tend not to use technologies that require preparation and much more time to organize.
- Lack of technical and theoretical knowledge is another barrier to using computer-assisted language learning technology. Not only is there a lack of knowledge about developing software to promote learning, but also there are many teachers who do not know how to use new technologies. In addition, little is known about the integration of these new means of learning into an overall plan, especially since some teachers consider CALL to be a new science, not knowing its theoretical revelation [14]. Therefore, improper use of technologies can negatively influence the teacher and the learner

2.3 Activities for Learners and Feedback

It is clear that the development of personal computers and the Internet has allowed CALL to take a new step and has shown enormous potential as a teaching tool [14].

Thanks to these special properties, which will be detailed later, the ALAO has been of great use to both the teacher and the learner.

With CALL socio-cognitive approach, language learning changes from learners' interaction with computers to interaction with other human beings through the computer, and students can communicate across languages and cultures [2]. Random access to web pages breaks the linear flow of instruction. By sending e-mail and attending discussion groups, learners can communicate with people they have never met. We can say that CALL allows a participant to share a message with a small group, the whole class, or an international debate, involving hundreds or thousands of people. Learners' access to the web will make them feel like they are participants in a global classroom and practicing global communication. This will allow them to escape "canned" knowledge and discover thousands of sources of information. As a result, their education will meet the need for interdisciplinary learning in a multicultural world [14].

2.4 Problem of Arab CALL

Despite the development of Arabic NLP in recent years, the number of tools made available to the ALAO to offer teachers and learners interactive environments for learning the Arabic language remains, from our point of view, very modest [1].

- "ArabVISL" is an interactive online tool for learning Arabic grammar. It allows Arabic learners to analyze sentences written in Arabic [6].
- "Arabic CALL" allows learners to produce sentences in different contexts. According to Shaalan, the system allows learners to recognize errors and guide them to self-correct [6]. The NLP resources used are: A grammar checking tool called GramCheck [8], a syntactic analyzer and an error analysis tool to check learners' responses.
- "SALA" allows Arabic learners to do derivation exercises according to schemes or conjugation activities. The feedback still remains classic True/False style [2].
- "rab-Learn" is a learning tool, including solutions from NLP. It is easy to use and allows Arabic teachers, non-NL specialists, to use resources and configure them in order to generate activities intended for learning [15]

To construct learning activities and evaluate them, to finely diagnose certain linguistic mechanisms, to provide learners with the appropriate remediations and the language information they need, NLP seems essential, since it involves manipulating in an almost automatic way of linguistic contents, statements, vocabulary elements, etc. [4, 5, 10].

In summary, there have been some Arabic learning tools, however, they all have common limitations in terms of variety of activities and quality and feedback. These systems are rigid since the data used is often static and defined in advance. The feedback in these tools does not adapt to the levels of the learners or the nature of the activities. The corrections are pre-established and they give no explanation as to the

nature of the error. All this pushed us to move towards the TELA Arabic language learning system.

3 Presentation of the "TELA" Environment

3.1 Functional Description

TELA (Toward Environnement Learning Arabic) is a learning environment including solutions from NLP [11, 16, 17]. It provides answers to questions considered fundamental for learning written Arabic. Thus, we have reaffirmed the important intersections between the research dynamics in CALL and research in the field of NLP [12]. It provides teachers and learners of the Arabic language with other NLP resources, methods and other tools necessary to boost learning [18]. These automatic language processing tools will be used in the generation of educational activities for teaching Arabic [10, 15, 19, 20].

The TELA environment shapes a set of "educational bricks", called scripts, which are modules which integrate NLP resources or treatments, and which each have a general educational objective. It offers a range of activities. The teacher can create his own exercises and make them available to his learners. He also has the choice of making them accessible and sharing them with other teachers. It constitutes the grains of the final educational objects called scenarios. A scenario is a succession of activities. Figure 1 represents the overall architecture of TELA. A description of its operation is detailed in the following paragraphs:

It is a learning tool, including solutions from NLP. It is easy to use and allows Arabic teachers to use resources and configure them in order to generate activities intended for learning.

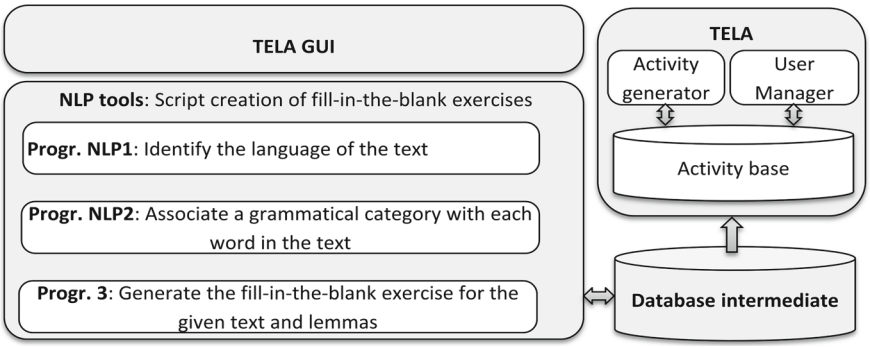


Fig. 1 Architecture of TELA

TELA is designed on several levels, associated with the database (or possibly a multifunction dictionary for certain types of conjugation, translation or derivation exercises), which make it possible to design scenarios:

- The function level, invisible to normal users, concerns NLP applications, such as text segmentation [12], labeling [19], spelling correction [18];
- The script level (internal view) is provided by the "lowest" part of the environment. It allows you to apply functions for language teaching. IT is transparent for TELA teacher-learners. Generate exercises for example, segment the text, assign a grammatical category and morphological information, generate derivation exercises etc.;
- The activity level (internal view of scripts) corresponds to what is traditionally designated as an exercise given to the learner to enable him or her to achieve a desired goal. The educational activities offered by TELA deal with particular phenomena and are of real educational interest. It involves working on a concept, revising a conjugation, where he must specify the gender, person and tense then he must select the type of instructions for this activity, etc. At this level, learning aids are also defined such as: a dictionary [13], a derivator [10], a conjugator [11], an Arabic–English translator [21], an Arabic–French translator [16] and the instructions and evaluation of exercise;
- The scenario level is a succession of activities ensuring personalized adaptation to the learner and offering new possibilities, with a specific educational goal, defined by the teacher.

3.2 Complete Cycle of TELA

The end user only receives from TELA the scenarios that he will be required to use according to the instructions of the teacher, author of each scenario. It is described in different main phases:

- Creation phase: In this phase the teacher creates educational modules using Arabic Tal tools.
- Orientation phase: To enable an educational path adapted to the learner's needs, the teacher defines in this phase the sequence of the different training entities which will be integrated into the learner's process. This sequence can be defined, either at the level of the educational modules, or in the definition of the learner's journey.
- Learning phase: In this part, the learner consults his educational modules, takes his evaluation tests and communicates with the other players on the platform. In this phase the teacher can collect feedback on the learner's learning session.
- Monitoring and evaluation phase: learner monitoring and evaluations are important elements in the training cycle. It is done during and at the end of the learning session. Indeed, educational monitoring allows teachers to know all the activities

carried out by the learner during their learning and to recover all the data on their activities, in order to analyze them.

Assessment tests allow teacher assessors to test and evaluate the understanding and knowledge of their learners, in order to know if the objective of their training is achieved.

3.3 Creating Scripts

Scripts are modules that can integrate NLP tools. For example, the automatic generation of each activity is considered a script, which requires parameterization by the teacher to achieve the desired educational content.

The creation of scripts is ensured by the "lowest" part of the system. It is transparent for TELA teacher-users who only perceive its results, namely the scripts themselves.

This module is essentially composed of NLP software from research laboratories or the commercial sector. Given the technical nature of the scripts and their levels of complexity for teachers who generally do not have sufficient knowledge of computers or NLP, ergonomic and simplified interfaces have been created. These interfaces make these tools invisible to language teachers while giving them the power to take full advantage of these resources.

Each of the programs can produce various results when applied to text. The chaining of two or more programs can lead to other results, generally richer in information. In this context, creating a script consists of defining the sequence of programs which, when applied to a text, leads to an educationally significant and usable result.

3.4 Feedback: Integration of Tools NLP in TELA

NLP is surely the missing area to allow a real qualitative evolution and especially an approach in the design of such systems. The integration of NLP tools into CALL offers virtual automation of activity creation.

The proposed approach is based on the integration of NLP tools, mainly our morphological analyzer TELAMA and our multifunction dictionary, into CALL systems for the automatic and semi-automatic creation of educational applications. An idea that will solve the problems of traditional online learning platforms. Learning languages is by no means an easy thing; educational activities must be carefully created to teach the desired concepts. For this, we provide teachers with a robust morphological analyzer of Arabic texts capable of adapting to the themes addressed by these activities.

The activities that we can design and carry out with CALL systems are varied. For TELA, we decided to offer interactive activities. With the NLP tool we have and other resources, we have the opportunity to offer teachers and learners an environment that promotes objective learning and a multitude of activities depending on the levels.

3.5 TELA: Learner View

From our personal analysis, we were able to see that the core component of our computer-assisted language learning system "TELA": case of the Arabic language, requires these following points so that the learner can work independently:

- He shares a message with a small group, the entire class, an associate class, or an international discussion list of hundreds or thousands of people.
- The learner in his learning by offering him varied activities, linked to the possibilities of the electronic medium, by making him more independent in relation to classroom activities. A typical CALT or CALL application contains multimedia content, exercises of different types and, possibly, grammar sheets specifying the material covered and managed automatically by the computer.
- The system offers different functionalities, the most classic of which are the management of groups of learners, the support of educational resources and the management of teacher/learner or learner/learner communications.
- The system allows learners to express their own words or sentences freely without following any rules to improve the efficiency of the system.

The main research perspectives that we have defined at the end of our work in the CALL part is to create a tool for managing courses, creating and distributing interactive courses, developing tasks and educational scenarios ensuring adaptation personalized to the learner and offering new possibilities: calculation of the following activity based on the learner's history, automatic generation of aids, automatic creation of educational reports, etc.

3.6 Assessment

Before the end of the development cycle of the TELA prototype, we carried out some experiments by having a teacher and a learner test the tool. This was on the one hand to check whether TELA provides teachers and learners with the most for which it was initially designed and on the other hand to take into consideration any comments during development. The proposed tool provides semi-automation appreciated by teachers, improvements in ergonomic aspects must be made.

4 Conclusion

In this article, we presented a computer environment, promoting lexical acquisition and helping to learn Arabic, based on NLP tools with unlimited vocabulary objectives and using them as simply as possible.

This objective is accomplished through effective resources, methods and approaches in the different parts of our research work. This desire to work is then the starting point for new opportunities, for the intersection of two fields of research, NLP and CALL, which have remained separated for a long time. In this research work, we make extensive use at the crossroads of three disciplines: language teaching, NLP and computer science.

References

1. Alkhudir, R.: Mobile assisted language learning in Saudi EFL classrooms: effectiveness, perception, and attitude. Theory Pract. Lang. Stud. **10**(2), 1620–1627 (2020). https://doi.org/10.17507/tpls.1012.16
2. Alghamdi, J., Holland, C.: A comparative analysis of policies, strategies and programmes for information and communication technology integration in education in the Kingdom of Saudi Arabia and the republic of Ireland. Educ. Inf. Technol. **25**, 4721–4745 (2020). https://doi.org/10.1007/s10639-020-10169-5
3. Alshumaimeri, Y., Gashan, A., Bamanger, E.: Virtual worlds for collaborative learning: Arab EFL learners' attitudes toward second life. World J. Educ. Technol.: Curr. Issues **11**(3), 198–204 (2019). https://doi.org/10.18844/wjet.v11i3.4235
4. Khalil, R., Mansour, A.E., Fadda, W.A., Almisnid, K., Aldamegh, M., Al-Nafeesah, A., Al-Wutayd, O.: The sudden transition to synchronized online learning during the COVID-19 pandemic in Saudi Arabia: a qualitative study exploring medical students' perspectives. BMC Med. Educ. **20**(1), 1–10 (2020). https://doi.org/10.1186/s12909-020-02208-z
5. Berry, S.: The role of video and text chat in a virtual classroom: how technology impacts community. In: Educational Technology and Resources for Synchronous Learning in Higher Education, pp. 173–187. IGI Global (2019). https://doi.org/10.4018/978-1-5225-7567-2.ch009
6. Dichy. J., Farghaly. A.: Roots & Patterns vs. stems more grammar-lexis specifications: on what basis should a multilingual lexical database centerd on Arabic be built? (2003)
7. Bacha, K.: Contribution to a new approach to analyzing Arabic words. ICCCI **2**, 46–57 (2019)
8. Almogheerah, A.: Exploring the effect of using WhatsApp on Saudi female EFL students' idiom-learning. Arab. World Engl. J. (AWEJ) **11**(4), 328–350 (2020). https://doi.org/10.24093/awej/vol11no4.22
9. Khemakhem A., Gargouri B., Ben Hamadou, A., Francopoulo G.: ISO standard modeling of a large Arabic dictionary. J. Nat. Lang. Eng. (2015)
10. Bacha, K.: Machine translation system on the pair of Arabic/English. In: International Conference on Knowledge Engineering and Ontology Development. KEOD, Barcelona, Spain (2012)
11. Bacha, K.: Toward a model of statistical machine translation Arabic-French. In: International Conference on Advanced Learning Technologies and Education. ICALTE, Hammamet, Tunisia (2014)
12. Bacha, K.: Contribution to the achievement of a spellchecker for Arabic. In: 17th International Conference on Intelligent Text Processing and Computational Linguistics (CICLing). Konya, Turkey (2016)

13. Bacha, K.: TELA: Toward Environmental Learning Arabic. In: The International Conference on Artificial Intelligence (CIIA 11). WORLDCOMP 11, Las Vegas, Nevada, USA, p. 6 (2011)
14. Ben Mohamed M.: Thèse: Intégration des techniques de TAL dans l'apprentissage assisté par ordinateur pour l'enseignement de la langue arabe Sfax (2016)
15. Zribi, C.B.O., Torjmen, A., Ahed, M. B.: A multi-agent system for pos-tagging vocalized Arabic texts. Int. Arab. J. Inf. Technol. **4**(4). RIADI Laboratory, University of the Manouba, Tunisia (2007)
16. Bacha, K.: Designing a model of Arabic derivation, for use in computer assisted teaching. In: International Conference on Knowledge Engineering and Ontology Development. KEOD, Barcelona, Spain (2012)
17. Bacha, K.: Design and implementation of a model of segmentation environmental computer assisted learning "TELA". CIIA'sOusse, Tunisia (2014)
18. Bacha K.: Designing a model combination of Arabic, for use in computer assisted teaching. In: World Congress on Computer Applications and Information Systems, pp. 1–7 (2014)
19. Bacha, K.: Morphological analysis in the environment "TELA". Procedia Comput. Sci. **62**, 191–215. Elsevier, SCSE 2015, California, USA (2015)
20. Bacha, K., Jemni, M.: Toward a learning system based on Arabic NLP tools. Int. J. Inf. Retr. Res. (IJIRR) **6**(4), 15 (2016)
21. Bacha, K.: Design of a synthesizer and a semantic analyzer's multi Arabic, for use in computer assisted teaching. Int. J. Inf. Sci. Appl. (IJISA), 11–33 (2012)
22. Bacha, K.: Help system for creating educational resources for Arabic. Int. J. Knowl. Syst. Sci. **9**(3), 32–47 (2018)
23. Alswilem, D.A.A.M.: Saudi English teachers' use of technology in secondary classrooms: perceptions, barriers, and suggestions for improvement. Adv. Lang. Lit. Stud. **10**(6), 168–178 (2019). https://doi.org/10.7575/aiac.alls.v.10n.6p.168
24. Han, Y.: Connecting the past to the future of computer-assisted language learning: theory, practice and research. Issues Trends Learn. Technol. **8**(1), 1–13 (2020). https://doi.org/10.2458/azu_itlt_v8i1_han

AI in Adaptive Learning: Challenges and Opportunities

Aicha Er-Rafyg, Hajar Zankadi, and Abdellah Idrissi

Abstract The convergence of adaptive learning and artificial intelligence holds significant promise for reshaping education. This article explores the role of AI in adaptive learning, examining the impact on teaching and learning paradigms. It thoroughly explores essential facets of employing AI tools in adaptive learning, clarifying both the opportunities and challenges essential in this integration. By addressing key aspects of AI's potential benefits and obstacles, the paper provides a comprehensive perspective, emphasizing ethical considerations and the inclusivity of AI integration in adaptive learning.

Keywords Artificial intelligence · Adaptive learning · Personalized learning · Artificial intelligence in education · AIED

1 Introduction

The intersection of artificial intelligence (AI) and adaptive learning is a significant development in the rapidly changing field of education, with the potential to completely transform the way we approach adaptive learning [1]. Adaptive learning, a dynamic process adjusting instructional content based on individual learner comprehension [2], has demonstrated its potential to offer diverse pedagogical benefits, ranging from acceleration to fostering metacognition and interactive learning.

The integration of AI technologies, ranging from data analytics to machine learning algorithms, will enhance the design and functionality of adaptive e-learning

A. Er-Rafyg (✉) · H. Zankadi · A. Idrissi
Artificial Intelligence & Data Science Group, IPSS Team, Faculty of Science, Mohammed V University in Rabat, Rabat, Morocco
e-mail: aicha_errafyg@um5.ac.ma

H. Zankadi
e-mail: hajar_zankadi@um5.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_26

systems. This integration brings forth the promise of providing personalized and tailored learning experiences that dynamically adapt to the unique characteristics, preferences, and needs of each learner. AI is bringing new pedagogical approaches to education that are beneficial to both teachers and learners as it covers numerous fields. Teachers may focus on more critical activities, while learners benefit from creative and varied learning methods. The incorporation of AI technology improves the overallquality of learning experiences.

In this paper, we explore how adaptive learning and AI can benefit each other, examining the challenges and opportunities that come with their collaboration. The aim behind this work is to understand the technical difficulties, teaching aspects, and ethical issues involved in bringing AI into adaptive learning for more personalized and effective education.

The remainder sections of this paper are organized as follows: Sect. 2 highlights an overview of adaptive learning with examples of developed systems. Section 3 points out related work regarding the use of AI in adaptive learning. Moreover, Sects. 4 and 5 present the opportunities and the challenges that surround the integration of AI technology in adaptive learning while Sect. 6 highlight the conclusion and future work.

2 Adaptive Learning

2.1 Overview of Adaptive Learning

Adaptive learning, a dynamic process adjusting instructional content based on learner comprehension, offers diverse pedagogical benefits, including acceleration, remediation, metacognition, and interactive learning [3]. Its implementation involves adaptive web applications, systems, and design frameworks, utilizing data analytics and machine learning algorithms [4].

Practically, adaptive learning is a personalized approach delivering customized learning experiences, adapting to individual needs and employing assessments for valuable feedback [5, 6]. The benefits encompass improved learner engagement, increased retention, and metacognitive skill development. Challenges include the need for high-quality data, potential bias, and instructor training [7].

Positioned as an educational technology, adaptive learning employs customized support, aiming to benefit diverse stakeholders. For teachers, it enhances efficiency by providing tools for personalized teaching, saving time, and offering insightful data [8]. Simultaneously, it facilitates personalized instruction for learners, fostering potential through tailored, individualized guidance [9]. Addressing fiscal challenges, it offers cost-effective options, contributing to increased pass rates and accelerated learner proficiency.

The design and functionality of adaptive e-learning systems depend significantly on their architecture. As highlighted in [10], the architecture of any adaptive learning system primarily relies on three fundamental models [11]: the learner model, the domain model, and the adaptation model (Fig. 1). The following provides a description of each model:

1. **Learner Model**: This component provides a structured presentation of the learner's characteristics, including objectives, knowledge background, preferences, learning styles, reasoning styles, experiences, physical skills, emotions, needs, habits, motivation, culture, personality, interests, cognitive styles, and social context [12].
2. **Adaptation Model**: This model is responsible for managing the adaptation process, including the rules and algorithms used to personalize the learning experience based on the learner's characteristics and the domain model [13].
3. **Domain Model**: This model represents the knowledge domain and the instructional content. It includes the structure of the learning material, the relationships between different concepts, and the prerequisites for understanding specific topics [14].

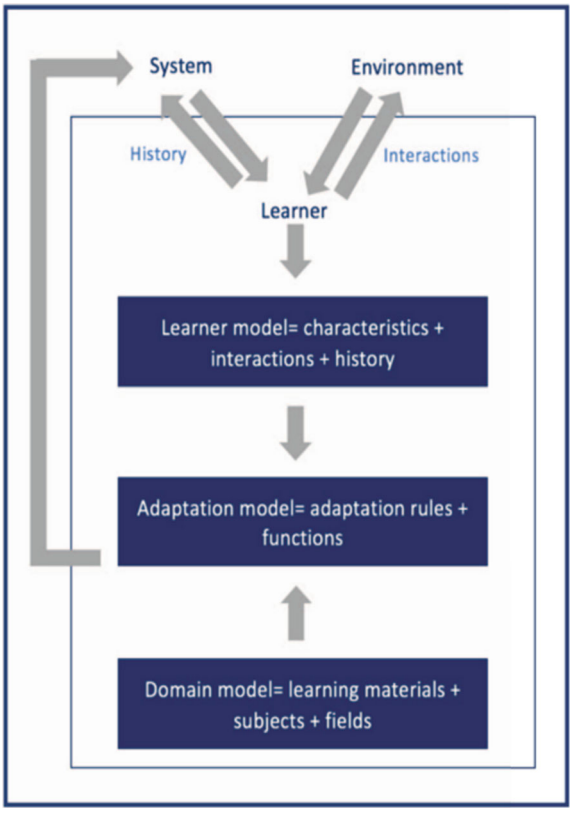


Fig. 1 Adaptive learning systems' components [10]

These components work together to create a personalized learning experience for each individual user, taking into account their unique characteristics, preferences, and needs. The architecture of adaptive e-learning systems is designed to provide a dynamic and tailored learning environment that adapts to the specific requirements of each learner.

2.2 Examples of Adaptive Learning Systems

Adaptive learning systems, designed in various iterations to cater to specific educational needs, collectively contribute to the overarching goal of transforming and enriching personalized learning experiences. These technological solutions, exemplified by platforms such as Khan Academy [1, 15], Smart Sparrow [16], and DreamBox [17], share a common objective: to enhance the efficiency of educational processes.

For instance, Khan Academy, a widely recognized adaptive learning platform, personalizes lessons in subjects like mathematics and science, offering targeted exercises and assessments to address individual learning gaps with precision. The primary aim is to save teachers valuable time while providing insightful data on each student's progress and learning capabilities.

Beyond teacher efficiency, adaptive learning endeavors to facilitate personalized instruction for students, allowing each individual to receive guidance and practice tailored to their capabilities. Platforms like Duolingo [1] and ALEKS [18, 19] demonstrate this goal by utilizing adaptive algorithms to customize language lessons or assess a student's knowledge and adapt the curriculum in areas requiring improvement, respectively. These applications, through scaffolding techniques, provide individualized instruction, fostering the development of students' potential within and beyond the classroom, all while delivering immediate feedback based on their strengths and weaknesses.

In addressing the fiscal challenges faced by school administrators, adaptive learning emerges as a practical solution. Platforms such as McGraw-Hill [20] Connect and Adaptive Learning Platforms in Higher Education offer cost-effective options and contribute to increased pass rates, reduced fail rates, and accelerated student proficiency. Moreover, adaptive learning serves as an edtech tool that empowers teachers and staff, offering complete ownership in content development and adaptation to specific student needs, aligning with the insights of Aleven et al. [8]. These examples collectively highlight the multifaceted goals of adaptive learning, encapsulating efficiency enhancement for educators, personalized instruction for students, fiscal efficacy for administrators, and empowerment for educational practitioners, underscoring its transformative potential in the realm of education.

3 Related Work: AI in Adaptive Learning

The integration of artificial intelligence (AI) in education area has resulted in a diversity of pedagogical approaches for teachers and learners. This incorporation of technology has given teachers the ability to automate repetitive tasks and allowing them to focus on moreimportant activities. On the other hand, it allows learnersto learn in a variety and creative ways. AI in education emphasizes intelligent learningand teaching assistance. Technologies like image and face recognition andnatural language processing transform learning practices by enhancing both the learning efficiencyand learner experiences. Particularly, the integration of AItechnology along with adaptive learningleads to improvement in the learning quality.

One of the forms of AI based adaptive learning system is the use of ChatGPT. It is an AI powered tool that can facilitate access to information for learners and can adjust the teaching methods based on a learner's progress and performance [21]. Many works have highlighted the use of ChatGPT and its challenges when used for educational purposes in general and for adaptive learning in particular [22, 23] and [24].

Authors in [25] analyzed 147 studies from 2014 to 2020 to identify AI interventions and analytical methods. The purpose is to provide insights for designing effective AI-enabled adaptive learning systems to address specific learning challenges. Similarly, authors in [26], delves into AI in adaptive learning with a focus on English Language Teaching, examining the concepts, uses, and evaluations of adaptive tools. The findings emphasize the tools' value in complementing teaching, offering inclusive opportunities for tailored instruction, and empowering learners to take responsibility for their learning. In another work [27], authors proposed a Bayesian network approach, offering an intuitive means for educational stakeholders to independently assess AI-enabled adaptive learning systems (AI-ALS), providing insights into enhancing learners' problem-solving abilities.

The work presented in [28], pointed out an AI based adaptive learning system called 'Yixue Squirrel AI' that outperformed both traditional classroom and another adaptive learning platform in term of learners performance. In another work [29], authors developed an Artificial Intelligence-Enabled Intelligent Assistant (AIIA) for adaptive learning. The AIIA system utilizes advanced AI and NLP techniques to create an interactive learning platform, reducing cognitive load by providing easy access to information and personalized learning support.

Authors in [30], introduced an adaptive learning system architecture utilizing Artificial Neural Networks to construct a Learner Model, outperforming Knowledge Tracing in predicting student performance by modeling relationships between curriculum concepts.

Various studies, including the analysis of interventions and methods, underscore the potential of AI-enabled adaptive learning systems in addressing specific challenges and enhancing the quality of learning. Notably, the utilization of tools like ChatGPT and innovative AI-based systems, demonstrate the transformative impact

of AI on personalized and efficient learning environments. In the following sections, we will highlight some the opportunities and challenges related to the use of AI in adaptive learning.

4 Opportunities of AI in Adaptive Learning

The widespread integration of AI across various educational domains has undoubtedly influenced the personal and professional development of both teachers and students, presenting numerous opportunities.

4.1 Technical Opportunities

- **Education Administration**:

AI applications in education, serving diverse roles, significantly influence the efficiency of administrative and management functions within the educational sphere. Teachers now benefit from enhanced capabilities, allowing for more effective execution of administrative tasks, including grading and delivering feedback to learners [31].

- **Real-Time feedback and Assessment**:

AI-driven systems enable instant analysis and evaluation of learners' performance, providing immediate feedback on their understanding and progress. These systems utilize algorithms to assess responses, identify patterns, and offer personalized insights tailored to individual learning styles. The integration of AI enhances the speed and accuracy of assessment, allowing educators to adapt their teaching strategies in real-time based on the evolving needs of each learner. This empowers learners with targeted guidance, contributing to a more effective and personalized educational experience [32–34].

- **Dropt-out Prediction**: AI plays a pivotal role in dropout prediction within e-learning, utilizing advanced algorithms and data analysis to identify patterns indicative of potential learner disengagement or non-completion. This predictive capability allows for timely interventions and tailored support, addressing issues before they lead to dropout, and enhancing overall retention rates in e-learning environments [35–37].
- **Intelligent Tutoring Systems**: Intelligent Tutoring Systems (ITS) represent a sophisticated application of artificial intelligence in education. These systems go beyond traditional teaching methods by employing AI algorithms to deliver personalized and adaptive guidance to learners [38]. In essence, ITS act as virtual tutors, utilizing advanced technologies such as machine learning, natural language

processing, and data analytics to provide personalized assistance, answering questions, offering explanations, and guiding students through challenging concepts [39, 40].
- **Integration with Emerging Technologies**: Explores opportunities for AI in adaptive learning to integrate with emerging technologies like virtual reality, augmented reality, facial recognition and predictive analysis [41].

4.2 Pedagogical Opportunities

- **Personalization of Learning Paths**: Involves the utilization of AI to create customized learning paths tailored to the specific needs and characteristics of individual learners. Through advanced algorithms and data analytics, AI analyzes learners' preferences, strengths, and weaknesses to dynamically adjust the trajectory of their educational experience. This include individualized content delivery, adaptive skill development and real time feedback integration [32, 42, 43].
- **Adaptive Content Delivery**: Relies on AI to delivering content as per the comprehension level of a learner. Learners vary in their learning styles, interests, knowledge levels, personality types, and other contributing factors [44]. This includes individualized comprehension assessment, content customization algorithms, personalized learning styles, etc.
- **Facilitation of collaborative learning**: Uses AI to promote collaborative learning by forming groups based on complementary skills, fostering teamwork [45, 46].
- **Identifying and adapting to Learning Styles**: Leverages AI to analyze how learners best understand information and adapt teaching methods to match individual learning styles [47–50].
- **Gamification and Interactive learning**: Integrates AI to support the use of gamification elements, creating interactive and engaging learning experiences [51–53].

5 Challenges in Adaptive Learning with AI

5.1 Technical Challenges

- **Interoperability and integration**: Integrating AI systems with existing educational technologies and infrastructure can be complex. Ensuring seamless interoperability and integration with various platforms and tools is a technical hurdle [54, 55].
- **Scalability**: Implementing AI solutions at scale, especially in large educational institutions or across entire education systems, poses challenges related to computational resources, infrastructure, and managing increased data volumes [56].

- **User interface and experience**: Designing user-friendly interfaces and experiences that cater to the diverse needs of educators, learners, and administrators is a technical challenge. The usability of AI tools is crucial for their effective adoption [57].

5.2 Pedagogical Challenges

- **Teacher training and professional development**: Teachers may lack the necessary training, skills and professional development to effectively incorporate AI tools into their teaching methods [58–60]. Bridging this gap is essential for maximizing the benefits of AI in the classroom.
- **Balancing automation and human interaction**: It's critical to find the correct balance betweenAI-driven automation and human interaction.An over-reliance on AI could make instruction less individualized and social [61].
- **Promoting critical thinking**: AI can streamline certain tasks, but promoting critical thinking and problem-solving skills remains a challenge [57, 62]. Educators need to ensure that AI complements rather than interferesthe development of these essential skills.
- **Addressing Resistance to Technology**: Some educators, students, and parents may resist the integration of AI in education due to concerns about job displacement, privacy issues, or a preference for traditional teaching methods. Overcoming this resistance requires effective communication and education [63].

5.3 Ethical and Privacy Concerns

The ethical challenges and inherent risks associated with AI systems in education appear to create diverse ethical challenges, among them:

- **Data Privacy and Security**:

An essential ethical consideration in applying AI in education revolves around the privacy and security of learner data. Since AI often needs to collect and analyze personal information for personalized learning experiences, it is crucial to appropriately use this data. To address this, institutions should establish strong data protection policies that include clear procedures for obtaining, managing, and deleting learner data, ensuring its security to intended purposes [64].

- **Bias and Discrimination**:

The risk of AI algorithms reinforcing biases and promoting discrimination within the educational domain. Machine learning models learn from existing data, which may inherently contain societal prejudices. If these biases become ingrained in the AI

systems implemented in classrooms, it could result in unjust treatment or restrictions for specific learner groups [32, 65].

- **Transparency and Explainability**:

AI algorithms frequently function as opaque systems, posing difficulties in interpreting their decision-making mechanisms. In the field of education, preserving transparency and explainability is vital to foster trust and accountability. It is essential for teachers, learners, and educational institutions to understand the rationale behind the recommendations or decisions generated by AI-driven systems, preventing uncritical acceptance of their outputs [66, 67].

- **Learner Profiling and Manipulation**:

AI-driven educational systems possess the capability to construct comprehensive student profiles, encompassing learning preferences, strengths, weaknesses, and behavioral patterns. Although these profiles can enhance personalized learning, there exists a concern regarding the potential manipulation or unauthorized utilization of such information. Educational institutions are advised to define the explicit purpose and extent of data collection, assuring that learner profiles are solely employed to enhance educational outcomes and not for exploitative purposes [68].

- **Equity and Access**:

A pivotal ethical concern involves guaranteeing equitable access to AI-driven educational technologies. Despite the potential of AI to customize education for individual needs, there is anobvious risk that current inequalities will increase. Learners from socioeconomically disadvantaged backgrounds may become even more marginalized as a result of differences in technology access and resources [69].

6 Conclusion and Future Work

The integration of AI in adaptive learning systems presents multifaceted opportunities and challenges across technical, pedagogical, and ethical dimensions. Technical opportunities include streamlined administration, real-time feedback, dropout prediction, and intelligent tutoring systems. Pedagogical opportunities encompass personalized learning paths, adaptive content delivery, collaborative learning facilitation, and identification of learning styles. However, challenges such as interoperability, scalability, user interface design, and continuous adaptation exist on the technical front. Pedagogical challenges involve teacher training, balancing automation and human interaction, promoting critical thinking, and addressing resistance to technology. Ethical concerns revolve around data privacy, bias, transparency, learner profiling, and equitable access.

For future work, further research can delve into addressing technical challenges by developing seamless integration solutions, scalable AI models, user-friendly interfaces, and adaptable systems. Pedagogically, emphasis on teacher training, refining

the balance between automation and human interaction, fostering critical thinking, and overcoming resistance to technology should be explored. Ethical considerations could be addressed through the development of comprehensive data privacy policies, bias mitigation strategies, transparent AI algorithms, and guidelines for responsible learner profiling. Additionally, investigations into ensuring equitable access and minimizing educational disparities could be a focal point for future research. In addition, various techniques [70–85] would be applied in this context to improve the system.

References

1. Gurramkonda, B., Sahaji, I.: AI in education: delving the transformative potential of AI-enhanced learning paths for the enrichment of student knowledge (2023)
2. Xie, H., Chu, H.-C., Hwang, G.-J., Wang, C.-C.: Trends and development in technology-enhanced adaptive/personalized learning: a systematic review of journal publications from 2007 to 2017. Comput. Educ. **140**, 103599 (2019). https://doi.org/10.1016/j.compedu.2019.103599
3. Hamal, O., El Faddouli, N.-E., Harouni, M.H.A., Lu, J.: Artificial intelligent in education. Sustainability **14**(5), 2862 (2022). https://doi.org/10.3390/su14052862
4. Imhof, C., Bergamin, P., McGarrity, S.: Implementation of adaptive learning systems: current state and potential. In: Isaias, P., Sampson, D.G., Ifenthaler, D. (Eds.) Online Teaching and Learning in Higher Education. In: Cognition and Exploratory Learning in the Digital Age. Cham: Springer International Publishing, pp. 93–115 (2020). https://doi.org/10.1007/978-3-030-48190-2_6
5. Moleka, P.: Exploring the role of artificial intelligence in education 6.0: enhancing personalized learning and adaptive pedagogy. Social Sciences, Preprint, Sep 2023. https://doi.org/10.20944/preprints202309.0562.v1
6. Osadcha, K., Osadchyi, V., Semerikov, S., Chemerys, H., Chorna, A.: The review of the adaptive learning systems for the formation of individual educational trajectory (2020). https://doi.org/10.31812/123456789/4130
7. ASCILITE 2004: Jones and Jo—ubiquitous learning environment—teaching system using ubiquitous technology. Accessed 29 Nov 2023. [Online]. Available: https://www.ascilite.org/conferences/perth04/procs/jones.html
8. Aleven, V., McLaughlin, E.A., Glenn, R.A., Koedinger, K.R.: Instruction Based on Adaptive Learning Technologies
9. Muñoz, J.L.R., et al.: Systematic review of adaptive learning technology for learning in higher education. J. Educ. Res. (2022)
10. Ennouamani, S., Mahani, Z.: An overview of adaptive e-learning systems, p. 347 (2017). https://doi.org/10.1109/INTELCIS.2017.8260060
11. Park, O., Lee, J.: Adaptive instructional systems. In: Handbook of research on educational communications and technology, 2nd edn. Routledge (2004)
12. Aissaoui, O.E., Oughdir, L.: A learning style-based ontology matching to enhance learning resources recommendation. In: 2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET), pp. 1–7 (2020). https://doi.org/10.1109/IRASET48871.2020.9092142
13. Pawlowski, J.M.: The quality adaptation model: adaptation and adoption of the quality standard ISO/IEC 19796–1 for learning, education, and training. J. Educ. Technol. Soc. **10**(2), 3–16 (2007)
14. Li, F., He, Y., Xue, Q.: Progress, challenges and countermeasures of adaptive learning: a systematic review. Educ. Technol. Soc. **24**(3), 238–255 (2021)

15. Pardos, Z.A, Tang, M., Anastasopoulos, I., Sheel, S.K., Zhang, E.: OATutor: An open-source adaptive tutoring system and curated content library for learning sciences research. In: Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems, pp. 1–17. Hamburg Germany: ACM (2023). https://doi.org/10.1145/3544548.3581574
16. Design of Adaptive Training Control in Dispatcher Training Simulators, IEEE Conference Publication, IEEE Xplore. Accessed Nov 28 2023. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10135704
17. Vasyliuk, A., Basyuk, T., Lytvyn, V.: Design and implementation of a Ukrainian-language educational platform for learning programming languages (2023)
18. Serhan, D., Welcome, N.: Adaptive learning: students' perceptions of equity and inclusion. Presented at the Society for Information Technology & Teacher Education International Conference, Association for the Advancement of Computing in Education (AACE), pp. 428–433 (2023). Accessed 28 Nov 2023. [Online]. Available: https://www.learntechlib.org/primary/p/221894/
19. Lim, L., Lim, S.H., Lim, W.Y.R.: Efficacy of an adaptive learning system on course scores Systems **11**(1), Art. no. 1 (2023) https://doi.org/10.3390/systems11010031
20. Maier, M, Ruder, P.: Teaching Principles of Microeconomics. Edward Elgar Publishing (2023)
21. Strzelecki, A.: To use or not to use ChatGPT in higher education? A study of students' acceptance and use of technology. Interact. Learn. Environ., 1–14 (2023) https://doi.org/10.1080/10494820.2023.2209881
22. Opara, E., Mfon-Ette Theresa, A., Aduke, T.C.: ChatGPT for Teaching, Learning and Research: Prospects and Challenges. Rochester, NY, Mar 01 2023. Accessed 05 Dec 2023. [Online]. Available: https://papers.ssrn.com/abstract=4375470
23. Elbanna, S., Armstrong, L.: Exploring the integration of ChatGPT in education: adapting for the future. Manag. Sustain. Arab Rev., vol. ahead-of-print, no. ahead-of-print (2023). https://doi.org/10.1108/MSAR-03-2023-0016
24. Kabudi, T.: Towards designing AI-enabled adaptive learning systems. Doctoral thesis, University of Agder (2023). Accessed 05 Dec 2023. [Online]. Available: https://uia.brage.unit.no/uia-xmlui/handle/11250/3062984
25. Kabudi, T., Pappas, I., Olsen, D.H.: AI-enabled adaptive learning systems: a systematic mapping of the literature. Comput. Educ. Artif. Intell. **2**, 100017 (2021). https://doi.org/10.1016/j.caeai.2021.100017
26. Delgado, H.O.K., de Azevedo Fay, A., Sebastiany, M.J., Silva, A.D.C.: Artificial intelligence adaptive learning tools: the teaching of English in focus. BELT—Braz. Engl. Lang. Teach. J. **11**(2), e38749–e38749 (2020). https://doi.org/10.15448/2178-3640.2020.2.38749.
27. How, M.-L., Hung, W.L.D.: Educational stakeholders' independent evaluation of an artificial intelligence-enabled adaptive learning system using Bayesian network predictive simulations. Educ. Sci. **9**(2), Art. no. 2 (2019). https://doi.org/10.3390/educsci9020110
28. Cui, W., Xue, Z., Thai, K.-P.: Performance comparison of an ai-based adaptive learning system in China. In: 2018 Chinese Automation Congress (CAC), pp. 3170–3175 (2018). https://doi.org/10.1109/CAC.2018.8623327
29. Sajja, R., Sermet, Y., Cikmaz, M., Cwiertny, D., Demir, I.: Artificial intelligence-enabled intelligent assistant for personalized and adaptive learning in higher education (2023). arXiv: https://doi.org/10.48550/arXiv.2309.10892
30. Chaplot, D.S., Rhim, E., Kim, J.: Personalized adaptive learning using neural networks. In: Proceedings of the Third (2016) ACM Conference on Learning @ Scale, in L@S '16, pp. 165–168. New York, NY, USA: Association for Computing Machinery, avril 2016. https://doi.org/10.1145/2876034.2893397
31. Chen, L., Chen, P., Lin, Z.: Artificial intelligence in education: a review. IEEE Access **8**, 75264–75278 (2020). https://doi.org/10.1109/ACCESS.2020.2988510
32. Akgun, S., Greenhow, C.: Artificial intelligence in education: addressing ethical challenges in K-12 settings. AI Ethics **2**(3), 431–440 (2022). https://doi.org/10.1007/s43681-021-00096-7
33. Luckin, R.: Towards artificial intelligence-based assessment systems. Nat. Hum. Behav. **1**(3), Art. no. 3 (2017). https://doi.org/10.1038/s41562-016-0028

34. Murphy, R.F.: Artificial Intelligence Applications to Support K–12 Teachers and Teaching: A Review of Promising Applications, Challenges, and Risks. RAND Corporation (2019). Accessed 12 Dec 2023. [Online]. Available: https://www.rand.org/pubs/perspectives/PE315.html
35. Tamada, M.M., de Magalhães Netto, J.F., de Lima, D.P.R.: Predicting and reducing dropout in virtual learning using machine learning techniques: a systematic review. Presented at the 2019 IEEE Frontiers in Education Conference (FIE), pp. 1–9. IEEE (2019)
36. Alsolami, F.: A hybrid approach for dropout prediction of MOOC students using machine learning. Int. J. Comput. Sci. Netw. Secur. **20**, 54–63 (2020)
37. Cobos, R., Olmos, L.: A learning analytics tool for predictive modeling of dropout and certificate acquisition on MOOCs for professional learning. Presented at the 2018 IEEE international conference on industrial engineering and engineering management (IEEM), pp. 1533–1537. IEEE (2018)
38. Mousavinasab, E., Zarifsanaiey, N., Niakan Kalhori, S.R., Rakhshan, M., Keikha, L., Ghazi Saeedi, M.: Intelligent tutoring systems: a systematic review of characteristics, applications, and evaluation methods. Interact. Learn. Environ. **29**(1), 142–163 (2021)
39. Hwang, G.-J., Xie, H., Wah, B.W., Gašević, D.: Vision, challenges, roles and research issues of Artificial intelligence in education. Comput. Educ. Artif. Intell. **1**, 100001 (2020). https://doi.org/10.1016/j.caeai.2020.100001
40. Guo, L., Wang, D., Gu, F., Li, Y., Wang, Y., Zhou, R.: Evolution and trends in intelligent tutoring systems research: a multidisciplinary and scientometric view. Asia Pac. Educ. Rev. **22**(3), 441–461 (2021)
41. The-Institute-for-Ethical-AI-in-Educations-Interim-Report-Towards-a-Shared-Vision-of-Ethical-AI-in-Education.pdf. Accessed 12 Dec 2023. [Online]. Available: https://www.buckingham.ac.uk/wp-content/uploads/2020/02/The-Institute-for-Ethical-AI-in-Educations-Interim-Report-Towards-a-Shared-Vision-of-Ethical-AI-in-Education.pdf
42. Hwang, G.-J.: Definition, framework and research issues of smart learning environments-a context-aware ubiquitous learning perspective. Smart Learn. Environ. **1**(1), 1–14 (2014)
43. Pratama, M.P., Sampelolo, R., Lura, H.: Revolutionizing education: harnessing the power of artificial intelligence for personalized learning. KLASIKAL J. Educ. Lang. Teach. Sci. **5**(2), Art. no. 2 (2023). https://doi.org/10.52208/klasikal.v5i2.877
44. Murtaza, M., Ahmed, Y., Shamsi, J.A., Sherwani, F., Usman, M.: AI-based personalized e-learning systems: issues, challenges, and solutions. IEEE Access **10**, 81323–81342 (2022). https://doi.org/10.1109/ACCESS.2022.3193938
45. Nye, B.D.: Intelligent tutoring systems by and for the developing world: a review of trends and approaches for educational technology in a global context. Int. J. Artif. Intell. Educ. **25**(2), 177–203 (2015). https://doi.org/10.1007/s40593-014-0028-6
46. Sayed, W.S., et al.: AI-based adaptive personalized content presentation and exercises navigation for an effective and engaging E-learning platform. Multimed. Tools Appl. **82**(3), 3303–3333 (2023). https://doi.org/10.1007/s11042-022-13076-8
47. Rasheed, F., Wahid, A.: Learning style detection in E-learning systems using machine learning techniques. Expert Syst. Appl. **174**, 114774 (2021). https://doi.org/10.1016/j.eswa.2021.114774
48. El Fazazi, H., Samadi, A., Qbadou, M., Mansouri, K., Elgarej, M.: A learning style identification approach in adaptive e-learning system. In: Information Systems and Technologies to Support Learning. In: Rocha, Á., Serrhini, M. (Eds.) Smart Innovation, Systems and Technologies, pp. 82–89. Cham: Springer International Publishing (2019). https://doi.org/10.1007/978-3-030-03577-8_10
49. Azzi, I., Jeghal, A., Radouane, A., Yahyaouy, A., Tairi, H.: A robust classification to predict learning styles in adaptive e-learning systems. Educ. Inf. Technol. **25**(1), 437–448 (2020). https://doi.org/10.1007/s10639-019-09956-6
50. Atlas, S.: ChatGPT for higher education and professional development: a guide to conversational AI. Coll. Bus. Fac. Publ. (2023) [Online]. Available: https://digitalcommons.uri.edu/cba_facpubs/548

51. Bennani, S., Maalel, A., Ben Ghezala, H.: Adaptive gamification in e-learning: a literature review and future challenges. Comput. Appl. Eng. Educ. **30**(2), 628–642 (2022). https://doi.org/10.1002/cae.22477
52. Kumar, A.: Gamification in training with next generation AI- virtual reality, animation design and immersive technology. J. Exp. Theor. Artif. Intell., 1–14 (2022) https://doi.org/10.1080/0952813X.2022.2125080
53. Daghestani, L.F., Ibrahim, L.F., Al-Towirgi, R.S., Salman, H.A.: Adapting gamified learning systems using educational data mining techniques. Comput. Appl. Eng. Educ. **28**(3), 568–589 (2020)
54. Bittencourt, I.I., Costa, E., Silva, M., Soares, E.: A computational model for developing semantic web-based educational systems. Knowl.-Based Syst. **22**(4), 302–315 (2009) https://doi.org/10.1016/j.knosys.2009.02.012
55. Rane, N.: Integrating building information modelling (BIM) and artificial intelligence (AI) for smart construction schedule, cost, quality, and safety management: challenges and opportunities. Cost Qual. Saf. Manag. Chall. Oppor. (2023)
56. Krauss C., et al.: Best-of-Breed: service-oriented integration of artificial intelligence in interoperable educational ecosystems. In: Uden, L., Liberona, D. (Eds.) Learning Technology for Education Challenges, in Communications in Computer and Information Science, pp. 267–283. Cham: Springer Nature Switzerland (2023) https://doi.org/10.1007/978-3-031-34754-2_22
57. Kasneci, E., et al.: ChatGPT for good? On opportunities and challenges of large language models for education. Learn. Individ. Differ. **103**, 102274 (2023). https://doi.org/10.1016/j.lindif.2023.102274
58. Enhancing pre-service teachers' technological pedagogical content knowledge (TPACK): a mixed-method study|Educational technology research and development. Accessed 06 Dec 2023. [Online]. Available: https://doi.org/10.1007/s11423-019-09692-1
59. Häkkinen, P., Järvelä, S., Mäkitalo-Siegl, K., Ahonen, A., Näykki, P., Valtonen, T.: Preparing teacher-students for twenty-first-century learning practices (PREP 21): a framework for enhancing collaborative problem-solving and strategic learning skills. Teach. Teach. **23**(1), 25–41 (2017). https://doi.org/10.1080/13540602.2016.1203772
60. Technology-related knowledge, skills, and attitudes of pre- and in-service teachers: the current situation and emerging trends—ScienceDirect. Accessed 06 Dec 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0747563220303022?via%3Dihub
61. AI and education: guidance for policy-makers—UNESCO Bibliothèque Numérique. Accessed 12 Dec 2023. [Online]. Available: https://unesdoc.unesco.org/ark:/48223/pf0000376709
62. Darwin Rusdin, D., Mukminatien, N., Suryati, N., Laksmi, E.D., Marzuki: Critical thinking in the AI era: an exploration of EFL students' perceptions, benefits, and limitations. Cogent. Educ. **11**(1), 2290342 (2024). https://doi.org/10.1080/2331186X.2023.2290342
63. Treve, M.: What COVID-19 has introduced into education: challenges facing higher education institutions (HEIs). High. Educ. Pedagog. **6**(1), 212–227 (2021). https://doi.org/10.1080/23752696.2021.1951616
64. Ethical principles for artificial intelligence in education|Education and Information Technologies. Accessed 06 Dec 2023. [Online]. Available: https://link.springer.com/article/10.1007/s10639-022-11316-w
65. Howard, A., Borenstein, J.: The ugly truth about ourselves and our robot creations: the problem of bias and social inequity. Sci. Eng. Ethics **24**(5), 1521–1536 (2018). https://doi.org/10.1007/s11948-017-9975-2
66. Hassija, V., et al.: Interpreting black-box models: a review on explainable artificial intelligence. Cogn. Comput., pp. 1–30 (2023)
67. Chaudhry, M.A., Cukurova, M., Luckin, R.: A transparency index framework for AI in education. In: Rodrigo, M.M., Matsuda, N., Cristea, A.I., Dimitrova, V., (Eds.) Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners' and Doctoral Consortium. In: Lecture Notes in Computer Science, pp. 195–198. Cham: Springer International Publishing (2022). https://doi.org/10.1007/978-3-031-11647-6_33.

68. Ahmad, S.F., et al.: Impact of artificial intelligence on human loss in decision making, laziness and safety in education. Humanit. Soc. Sci. Commun. **10**(1), 1–14 (2023)
69. Kamalov, F., Santandreu Calonge, D., Gurrib, I.: New era of artificial intelligence in education: towards a sustainable multifaceted revolution. Sustainability **15**(16), 12451 (2023)
70. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15** (4), 4876–4886 (2020)
71. Elhandri, K., Idrissi, A.: Comparative study of Top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
72. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
73. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
74. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv preprint arXiv:1307.5910
75. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and Skyline for cloud services research and selection system. In: International conference on Big Data and Advanced Wireless technologies (2016)
76. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF, 107–116 (2006)
77. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
78. Idrissi, A, Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
79. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mobile Robot. Intell. Syst. **14**(3), 65–70 (2020)
80. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
81. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
82. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
83. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
84. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless technologies (2016)
85. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)

Advancements in Artificial Intelligence for Healthcare Systems: Enhancing Efficiency, Quality, and Patient Care

Abatal Ahmed, Anass Elachhab, and Elkaim Billah Mohammed

Abstract Artificial intelligence (AI) has transformed the healthcare landscape, opening up a world of possibilities for improving efficiency, enhancing care quality, and optimizing patient outcomes. This article delves into the various applications of AI in healthcare, with a focus on medical imaging, predictive analytics, and personalized medicine. We explore the potential benefits, challenges, and ethical considerations that arise from integrating AI into healthcare systems. Additionally, we present real-world examples and case studies to showcase the impact of AI algorithms on healthcare outcomes. The findings underscore AI's ability to improve diagnostic accuracy, streamline processes, and enable personalized treatment plans. This article contributes to the expanding body of knowledge on AI's role in healthcare, emphasizing the transformative potential of AI technologies to revolutionize healthcare systems.

Keywords Artificial intelligence · Healthcare systems · Efficiency · Quality · Patient care · Medical imaging · Predictive analytics

1 Introduction

Healthcare is constantly evolving, and advancements in artificial intelligence (AI) are playing a major role in this transformation. Integrating AI technologies into healthcare systems is becoming increasingly popular due to its ability to enhance

A. Ahmed (✉)
Faculty of Sciences and Techniques, Settat, Morocco
e-mail: a.abatal@uhp.ac.ma

A. Elachhab
National School of Applied Sciences, Chouaib Doukkali University, El Jadida, Morocco
e-mail: Elachhab.a@ucd.ac.ma

E. B. Mohammed
Faculty of Science, Chouaib Doukkali University, El Jadida, Morocco
e-mail: elkaim_billah.mohammed@ucd.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_27

efficiency, improve the quality of care, and optimize patient outcomes. This makes it an attractive prospect for stakeholders across the sector.

AI algorithms and technologies offer innovative solutions to complex healthcare challenges by analyzing large volumes of data, recognizing patterns, and making predictions or recommendations. These capabilities have the potential to revolutionize healthcare delivery, transforming traditional practices and enabling a more personalized and efficient approach to patient care.

AI in healthcare is as multifaceted as the field of medicine itself. It is revolutionizing various domains such as medical imaging, predictive analytics, and personalized medicine.

In medical imaging, AI algorithms have shown remarkable performance in interpreting complex images, aiding in the detection and diagnosis of diseases such as cancer, cardiovascular conditions, and neurological disorders [1, 10]. By automating image analysis and providing accurate and timely insights, AI algorithms assist healthcare professionals in making more informed decisions and improving patient outcomes [23, 24].

AI is also transforming predictive analytics in healthcare. It can analyze vast amounts of data from electronic health records, wearable devices, and other sources to predict which patients are at risk of developing certain diseases or conditions [1, 25]. These predictions empower healthcare providers to take proactive measures, such as preventive treatments or lifestyle changes, potentially delaying or even preventing the onset of these conditions [1, 25].

In the realm of personalized medicine, AI is paving the way for customized treatment plans. It can tailor treatment decisions to individual patients based on their unique genetic, environmental, and lifestyle factors [1, 25]. This approach holds the promise of more effective treatments with fewer side effects, leading to improved patient care and outcomes [23, 24].

As AI continues to evolve, its impact on healthcare will only grow. We can expect to see even more innovative applications that further enhance the quality, efficiency, and personalization of healthcare delivery.

Integrating artificial intelligence (AI) into healthcare promises numerous advantages, but it also raises several concerns and ethical issues that require careful consideration. To achieve successful AI integration in healthcare, it is crucial to address data privacy and security, algorithmic bias, and the transparency and interpretability of AI algorithms. Moreover, regulatory frameworks and guidelines need to be established to govern the ethical use of AI in healthcare, ensuring fairness, equity, and patient-centered care.

To date, numerous studies and research works have investigated the applications of AI in healthcare systems. For instance, Smith et al. conducted a comprehensive survey on the use of AI in medical imaging, highlighting its potential to improve diagnostic accuracy and enhance radiologists' workflow [9]. Similarly, Rajkomar et al. developed a scalable machine learning model for predicting patient readmissions using electronic health records, showcasing the efficacy of AI in healthcare management [10].

AI algorithms and technologies offer innovative solutions to complex healthcare challenges by analyzing large volumes of data, recognizing patterns, and making predictions or recommendations. These capabilities have the potential to revolutionize healthcare delivery, transforming traditional practices and enabling a more personalized and efficient approach to patient care [22].

In this article, we delve into the transformative applications of AI in healthcare systems, exploring its potential to revolutionize medical imaging, predictive analytics, and personalized medicine. We discuss the potential benefits, challenges, and ethical considerations associated with AI integration, providing real-world examples to demonstrate its remarkable impact. Through a comprehensive analysis of relevant research and practical applications, we offer valuable insights into AI's transformative potential in enhancing healthcare systems and significantly improving patient care.

The structure of this article is organized into key sections. It start with an exploration of the Applications of AI in Healthcare, explore the transformative impact of AI in medical imaging, predictive analytics, and personalized medicine. Following this, the Integration of AI in Healthcare Systems is dissected, Explaining the fluidity assimilation of AI algorithms into existing healthcare frameworks. The subsequent section, Evaluation and Validation, critically assesses the performance and reliability of AI algorithms through rigorous testing methodologies. Collaborative Opportunities and Implications are then explored, shedding light on potential avenues for collaboration and the broader implications of the research. The ensuing section unravels Challenges and Ethical Considerations, addressing pertinent issues related to data privacy, security, and ethical use of AI in healthcare. Potential Benefits and Challenges are delineated to provide a comprehensive view, leading to the conclusive remarks in the final section.

2 Applications of AI in Healthcare

AI algorithms are being employed across various domains in healthcare, revolutionizing the way healthcare is delivered. In this section, we discuss three key areas where AI is making a significant impact: medical imaging, predictive analytics, and personalized medicine.

2.1 Medical Imaging

Medical imaging plays a critical role in diagnosis, treatment planning, and monitoring of diseases. AI algorithms, particularly deep learning algorithms, have demonstrated remarkable capabilities in interpreting medical images, such as X-rays, CT scans, and MRIs [11]. These algorithms can accurately detect abnormalities, including tumors,

lesions, and other anomalies, aiding radiologists and clinicians in making more accurate and timely diagnoses. By automating image analysis tasks, AI algorithms enhance efficiency, reduce interpretation errors, and contribute to improved patient outcomes. By automating image analysis tasks, AI algorithms enhance efficiency, reduce interpretation errors, and contribute to improved patient outcomes [11].

Algorithm 1 Enhanced AI algorithm for medical image analysis

Require: Medical image data
Ensure: Detected abnormalities
1: Preprocess the medical image data
2: Apply a trained deep learning model to segment regions of interest
3: Utilize advanced feature extraction techniques to capture fine-grained details
4: Incorporate contextual information and spatial relationships in the analysis
5: Perform multi-class classification to detect and classify abnormalities
6: Integrate uncertainty estimation to quantify confidence levels
7: Incorporate explainability methods to provide interpretable results
8: Implement ensemble learning to leverage multiple models' predictions
9: Enable real-time analysis for immediate feedback
10: Generate diagnostic report based on the detected abnormalities
11: **return** Detected abnormalities and diagnostic report

In this enhanced version of the AI Algorithm for Medical Image Analysis, several new features have been incorporated to further improve the algorithm's performance and capabilities:

- **Uncertainty Estimation**: The algorithm includes techniques to estimate uncertainty levels in its predictions. This provides an indication of the confidence and reliability of the algorithm's results, enabling better decision-making by healthcare professionals.
- **Explainability Methods**: The algorithm incorporates explainability methods to provide interpretable results. This allows healthcare professionals to understand the reasoning behind the algorithm's predictions, improving trust and aiding in clinical decision-making.
- **Ensemble Learning**: The algorithm leverages ensemble learning techniques, combining predictions from multiple models. This ensemble approach improves the robustness and accuracy of the algorithm's predictions, enhancing overall performance.
- **Real-Time Analysis**: The algorithm is designed to perform real-time analysis, enabling immediate feedback to healthcare professionals. This feature is particularly beneficial in time-sensitive scenarios, such as emergency situations, where quick and accurate analysis is crucial.

These new features enhance the algorithm's capabilities in terms of uncertainty estimation, interpretability, accuracy, and real-time analysis. They contribute to more reliable and actionable results, supporting healthcare professionals in making informed decisions based on the algorithm's outputs.

2.2 Predictive Analytics

Predictive analytics involves the use of AI algorithms to analyze large amounts of data, such as electronic health records, clinical notes, and demographic information, to identify patterns and make predictions about future events or outcomes. AI algorithms can identify risk factors, predict disease progression, and estimate patient outcomes. For example, predictive analytics algorithms have been used to predict hospital readmissions, enabling healthcare providers to intervene and provide targeted interventions to prevent readmissions [10]. By leveraging predictive analytics, healthcare systems can optimize resource allocation, improve care coordination, and enhance patient management. These predictive capabilities enable healthcare providers to allocate resources effectively, improve care coordination, and implement proactive measures to prevent adverse events [16].

Algorithm 2 Enhanced AI algorithm for predictive analytics

Require: Patient data (electronic health records, clinical notes, genetic data, etc.)
Ensure: Predicted outcomes or events
1: Preprocess and clean the patient data
2: Extract relevant features from the data, including demographic information, lab results, medical history, and genetic markers
3: Perform feature engineering to create derived features and capture complex relationships
4: Apply advanced data imputation techniques to handle missing values
5: Incorporate domain knowledge and expert insights into the feature selection process
6: Utilize machine learning algorithms, such as Random Forest, Support Vector Machines, or Gradient Boosting, to train predictive models
7: Optimize model hyperparameters using techniques like grid search or Bayesian optimization
8: Implement ensemble methods, such as model averaging or stacking, to combine multiple models and improve predictive performance
9: Perform cross-validation and robust evaluation to assess the generalization and reliability of the models
10: Monitor model performance over time and implement model retraining or updating mechanisms
11: Use the trained model to predict outcomes or events for new patient data
12: **return** Predicted outcomes or events

In this enhanced version of the AI Algorithm for Predictive Analytics, several new features have been incorporated to improve the predictive performance and robustness of the model:

- Additional Data Sources: The algorithm now incorporates diverse data sources, including genetic data, to capture a more comprehensive view of the patient's health profile and potential risk factors.
- Feature Engineering: Feature engineering techniques have been introduced to create derived features that capture complex relationships within the data. This helps to extract more meaningful information and improve the predictive power of the model.

- Advanced Data Imputation: The algorithm now includes advanced techniques for handling missing values in the data, ensuring that valuable information is not lost and minimizing potential bias in the predictions.
- Domain Knowledge Integration: The feature selection process now incorporates domain knowledge and expert insights to guide the selection of relevant features. This helps to focus on the most informative variables and improves the interpretability of the model.
- Ensemble Methods: Ensemble methods are implemented to combine multiple models and harness their collective predictive power. This enhances the robustness and accuracy of the predictions by leveraging diverse modeling approaches.
- Model Optimization and Hyperparameter Tuning: Techniques such as grid search or Bayesian optimization are used to optimize the model hyperparameters, fine-tuning the model to achieve the best performance on the given dataset.
- Monitoring and Model Maintenance: The algorithm includes mechanisms for monitoring model performance over time and implementing model retraining or updating, ensuring that the model remains accurate and relevant as new data becomes available.

These additional features enhance the algorithm's ability to handle complex healthcare data, extract meaningful insights, and make accurate predictions. The algorithm now offers improved performance and reliability in the context of predictive analytics for healthcare systems.

2.3 Personalized Medicine

Personalized medicine aims to tailor medical treatment to individual patients based on their unique characteristics, such as genetic makeup, lifestyle, and medical history. AI algorithms play a crucial role in analyzing complex datasets and generating insights that enable personalized treatment strategies. By integrating genetic data, clinical data, and treatment outcomes, AI algorithms can identify optimal treatment plans for individual patients, maximizing treatment efficacy and minimizing adverse effects. Furthermore, AI algorithms can assist in drug discovery by analyzing large datasets and predicting the effectiveness of potential drug candidates [8]. Personalized medicine, powered by AI, has the potential to revolutionize healthcare by providing tailored treatments and improving patient outcomes. The integration of AI in personalized medicine enables more precise and effective healthcare interventions, leading to improved patient outcomes and reduced healthcare costs [17].

In this enhanced version of the AI Algorithm for Personalized Medicine, several new features have been incorporated to improve the precision and efficacy of treatment recommendations:

- Genomic Data Analysis: The algorithm now incorporates advanced genomic data analysis techniques to identify relevant biomarkers and genetic variants that can influence treatment response and potential adverse reactions.

Algorithm 3 AI algorithm for personalized medicine

Require: Patient data (genetic data, clinical data, etc.)
Ensure: Personalized treatment plans
1: Preprocess and integrate patient data from multiple sources
2: Analyze genetic data to identify relevant biomarkers and genetic variants
3: Use machine learning algorithms to analyze clinical data and treatment outcomes
4: Generate personalized treatment recommendations based on the analysis
5: **return** Personalized treatment plans

- Machine Learning for Treatment Response Prediction: Machine learning algorithms, such as Random Forest, Support Vector Machines, or Deep Neural Networks, are utilized to analyze patient data, including demographic information, medical history, genetic markers, and lifestyle factors, to predict individualized treatment responses.
- Recommendation Systems for Personalized Treatment Plans: The algorithm employs recommendation systems that integrate patient-specific characteristics and treatment-related factors to generate personalized treatment plans. It takes into account factors such as treatment effectiveness, potential side effects, drug-drug interactions, and patient preferences to provide tailored recommendations.
- Integration of Clinical Guidelines and Evidence-Based Practices: The algorithm incorporates established clinical guidelines and evidence-based practices into the decision-making process. It ensures that the personalized treatment plans align with the latest medical knowledge and best practices.
- Real-time Monitoring of Treatment Progress: The algorithm includes mechanisms for real-time monitoring of patient treatment progress, such as tracking biomarker levels or analyzing patient-reported outcomes. This allows for timely adjustments or modifications to the treatment plan based on emerging patient data.
- Continuous Learning and Updating: The algorithm has the capability to continuously learn from new patient data and update treatment recommendations over time. It adapts to evolving patient conditions and incorporates new scientific insights, ensuring that the personalized treatment plans remain up to date and relevant.

These enhanced features empower the AI Algorithm for Personalized Medicine to provide tailored treatment recommendations based on individual patient characteristics, including genomic information, medical history, and treatment response prediction. By leveraging these advanced techniques, the algorithm enables more precise and personalized medicine, ultimately improving patient outcomes, treatment effectiveness, and patient satisfaction.

3 Integration of AI in Healthcare Systems

This section provides an overview of the key components involved in integrating AI in healthcare systems to enhance efficiency, improve quality of care, and optimize patient outcomes. Figure 1 demonstrates the relationships between different elements

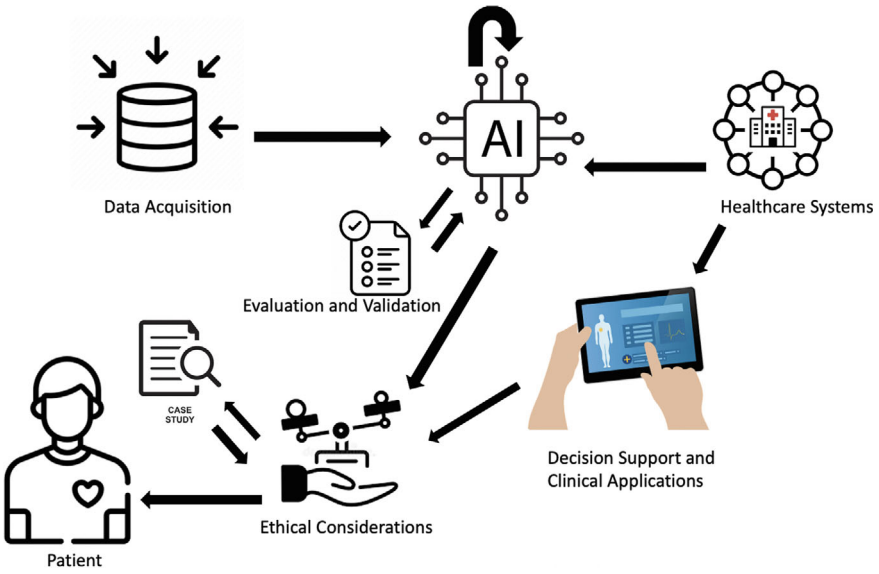


Fig. 1 Schema of AI integration in healthcare systems

and emphasizes the transformative potential of AI technologies in healthcare. The framework includes the following components:

- Data Acquisition: Collecting and curating diverse and high-quality healthcare data, including medical images, electronic health records, and genomic data.
- AI Algorithms and Models: Developing and deploying AI algorithms and models, such as deep learning models for medical image analysis, predictive analytics algorithms, and personalized medicine algorithms.
- Integration with Healthcare Systems: Integrating AI algorithms with existing healthcare systems, including electronic health record systems, radiology systems, and clinical decision support systems.
- Decision Support and Clinical Applications: Using AI algorithms to provide decision support for healthcare professionals, facilitate clinical decision-making, and enable personalized treatment plans.
- Evaluation and Validation: Assessing the performance and effectiveness of AI algorithms through rigorous evaluation and validation processes, including benchmarking against existing standards and comparing outcomes with traditional methods.
- Ethical Considerations: Addressing ethical considerations related to data privacy, security, algorithmic bias, and patient consent to ensure the responsible and ethical use of AI in healthcare systems.
- Real-world Examples and Case Studies: Presenting real-world examples and case studies that demonstrate the impact and effectiveness of AI algorithms in improving healthcare outcomes and patient care.

- Challenges and Future Directions: Discussing the challenges faced during AI integration in healthcare systems and highlighting potential future directions for research and development.

By presenting this framework, we can showcase the comprehensive approach taken to leverage AI in healthcare systems and emphasize the value and potential impact of our work.

4 Evaluation and Validation

To assess the impact and value of AI integration in healthcare systems, we employ a comprehensive evaluation framework, as depicted in Schema 2 (Fig. 2). This schema focuses on evaluating the effectiveness and benefits of AI algorithms in improving efficiency, quality of care, and patient outcomes. The framework includes the following components:

- Study Design: Describing the study design, including the study population, data sources, and study duration.
- Intervention: Detailing the AI algorithms or interventions implemented in the healthcare system, such as a deep learning model for medical image analysis or a predictive analytics algorithm for risk prediction.
- Outcome Measures: Identifying the outcome measures used to evaluate the impact of AI integration, such as reduction in diagnostic errors, improvement in treatment response prediction, or cost savings.
- Data Collection and Analysis: Explaining the data collection process, including the collection of pre-intervention and post-intervention data. Describing the statistical or analytical methods used to analyze the data and assess the impact of AI integration.

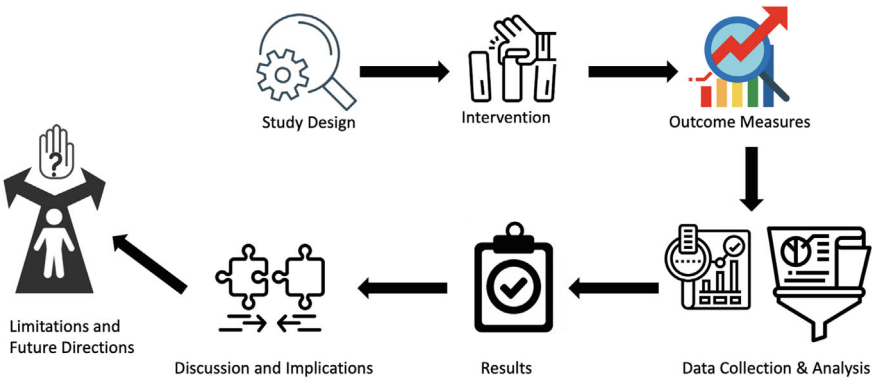


Fig. 2 Schema of impact evaluation of AI integration in healthcare systems

- Results: Presenting the results of the impact evaluation, including quantitative measures, statistical analysis, and any significant findings or improvements observed.
- Discussion and Implications: Discussing the implications of the results, including the potential benefits of AI integration in healthcare systems, the challenges encountered, and the generalizability of the findings.
- Limitations and Future Directions: Acknowledging the limitations of the study, such as sample size or data availability, and suggesting future directions for research and implementation.

By utilizing this schema, we can highlight the empirical evidence and quantitative assessment of the value and impact of AI integration in healthcare systems, further enhancing the significance of our work.

5 Collaborative Opportunities and Implications

Our research on leveraging artificial intelligence (AI) in healthcare systems presents significant opportunities for collaboration and implications for the broader scientific and healthcare communities. In this section, we outline potential collaborative avenues and highlight the implications of our findings.

5.1 Collaborative Opportunities

Our work opens doors for collaboration with professionals and researchers in the following areas:

- **Healthcare Professionals**: Collaborate with healthcare practitioners to implement and validate AI solutions in real-world clinical settings. Their insights are invaluable for refining algorithms and ensuring seamless integration into existing healthcare workflows.
- **Data Scientists and Machine Learning Experts**: Join forces with experts in data science and machine learning to advance algorithmic techniques, explore new models, and address challenges related to large-scale data processing and analysis.
- **Ethics and Policy Experts**: Collaborate with experts in ethics, policy, and healthcare regulations to navigate the complex landscape of responsible AI implementation. Addressing ethical considerations is crucial for ensuring patient privacy, fairness, and transparency.
- **Interdisciplinary Researchers**: Engage with researchers from diverse disciplines, such as computer science, medicine, and public health, to foster interdisciplinary approaches. Cross-disciplinary collaboration enhances the robustness and applicability of our AI-driven healthcare solutions.

5.2 Implications of Our Research

The implications of our research extend beyond the immediate scope and contribute to:

- **Enhanced Patient Care**: The integration of AI in healthcare systems enhances diagnostic accuracy, streamlines processes, and facilitates personalized treatment plans. Improved patient outcomes and experiences are direct implications of our work.
- **Scientific Advancement**: Our findings contribute to the broader scientific understanding of AI applications in healthcare. This includes advancements in medical imaging, predictive analytics, and personalized medicine.
- **Technological Innovation**: The development and deployment of AI algorithms in healthcare drive technological innovation. This has the potential to inspire new solutions, applications, and approaches in the broader field of artificial intelligence.
- **Policy and Regulation Development**: Our work underscores the need for robust policies and regulations governing the ethical use of AI in healthcare. This has implications for policymakers and regulatory bodies seeking to establish guidelines for responsible AI integration.

Collaboration in these areas and consideration of the broader implications of our work will contribute to the continued success and impact of our research in shaping the future of healthcare systems.

6 Challenges and Ethical Considerations

While AI holds immense promise for revolutionizing healthcare, its implementation presents a multitude of challenges and ethical considerations that must be carefully addressed. One of the primary hurdles lies in the need for vast, diverse, and high-quality datasets to train AI algorithms. Access to comprehensive and representative data is paramount to ensuring the accuracy, reliability, and generalizability of these models. Moreover, the integration of AI into healthcare systems raises critical concerns regarding data privacy, security, and patient consent. Robust measures must be implemented to safeguard patient data, ensure compliance with data protection regulations, and uphold patient confidentiality and trust.

Ethical considerations arise when using AI algorithms in healthcare. Transparency and interpretability of AI algorithms are crucial for building trust among healthcare professionals and patients. AI algorithms should be able to explain their decision-making processes and provide justifications for their predictions or recommendations. Furthermore, issues related to algorithmic bias, fairness, and equity must be carefully addressed to ensure that AI algorithms do not perpetuate existing healthcare disparities or discriminate against certain patient populations.

7 Potential Benefits and Challenges

There are many potential advantages to integrating AI in healthcare, but there are disadvantages as well, including ethical questions.

Improved Diagnosis and Treatment: AI can help medical professionals diagnose conditions more quickly and accurately, as well as suggest treatments. Patients may heal more quickly and have better results as a result of this.

8 Conclusion

This article delves into the ever-evolving landscape of AI in healthcare, exploring its diverse applications and integration into healthcare systems. It begins by examining AI's transformative impact in medical imaging, predictive analytics, and personalized medicine, highlighting its ability to improve diagnostic accuracy, streamline processes, and enable tailored treatment plans.

The article then delves into the seamless integration of AI into healthcare systems, emphasizing the harmonious coexistence of AI algorithms within the existing healthcare framework. It stresses the importance of rigorous evaluation and validation of AI algorithms to ensure their reliability and efficacy.

Considering the collaborative opportunities and implications, the article envisions potential partnerships and broader implications for the healthcare domain. It also acknowledges the challenges and ethical considerations surrounding data privacy, security, and ethical AI usage, emphasizing the need for responsible implementation.

By meticulously delineating potential benefits and challenges, the article provides a balanced perspective, highlighting both the promises and hurdles in integrating AI into healthcare. It concludes by encapsulating the strides made, the challenges recognized, and the potential future trajectories in this ever-evolving field.

Acknowledgements The author would like to acknowledge the support and contributions of colleagues in the field of healthcare technology and artificial intelligence.

References

1. Esteva, A., Kuprel, B., Novoa, R.A., Ko, J., Swetter, S.M., Blau, H.M., Thrun, S.: Dermatologist-level classification of skin cancer with deep neural networks. Nature **542**(7639), 115–118 (2017)
2. Ravi, D., Wong, C., Deligianni, F., Berthelot, M., Andreu-Perez, J., Lo, B.: Deep learning for health informatics. IEEE J. Biomed. Health Inform. **21**(1), 4–21 (2017)
3. Chartrand, G., Cheng, P.M., Vorontsov, E., Drozdzal, M., Turcotte, S., Pal, C.J., Tang, A.: Deep learning: a primer for radiologists. Radiographics **37**(7), 2113–2131 (2017)
4. Miotto, R., Wang, F., Wang, S.: Deep learning for healthcare: review, opportunities, and challenges. Briefings Bioinform. **19**(6), 1236–1246 (2018)

5. Rajpurkar, P., Irvin, J., Zhu, K., Yang, B., Mehta, H., Duan, T., Langlotz, C.P.: CheXNet: radiologist-level pneumonia detection on chest X-rays with deep learning (2017). arXiv preprint arXiv:1711.05225
6. Topol, E.J.: High-performance medicine: the convergence of human and artificial intelligence. Nat. Med. **25**(1), 44–56 (2019)
7. Choi, E., Bahadori, M.T., Schuetz, A., Stewart, W.F., Sun, J.: Doctor AI: predicting clinical events via recurrent neural networks. J. Biomed. Inform. **63**, 327–337 (2016)
8. Chen, X., Ishwaran, H., Eils, R.: Personalized therapy scheduling: algorithmic approaches. Briefings Bioinform. **21**(1), 100–111 (2020)
9. Smith, T.B., Smith, B.C., Ryan, P.: The use of artificial intelligence in medical imaging: a review. Radiography **26**(4), 355–362 (2020)
10. Rajkomar, A., Oren, E., Chen, K., Dai, A.M., Hajaj, N., Hardt, M., Ng, A.Y.: Scalable and accurate deep learning with electronic health records. NPJ Dig. Med. **1**(1), 1–10 (2018)
11. Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A., Ciompi, F., Ghafoorian, M., Sánchez, C.I.: A survey on deep learning in medical image analysis. Med. Image Anal. **42**, 60–88 (2017)
12. Gulshan, V., Peng, L., Coram, M., Stumpe, M.C., Wu, D., Narayanaswamy, A., Webster, D.R.: Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. JAMA **316**(22), 2402–2410 (2016)
13. Ma, J., Wu, F., Jiang, T., Jiang, J.: Applying deep learning in medical images: the state-of-the-art. Sci. China Inform. Sci. **61**(7), 070101 (2018)
14. Raghu, M., Zhang, C., Kleinberg, J., Bengio, S.: Transfusion: understanding transfer learning for medical imaging. Adv. Neural Inform. Process. Syst. **32**, 3347–3357 (2019)
15. Litjens, G., Ciompi, F., Setio, A.A.: A survey on deep learning in medical image analysis. Med. Image Anal. Clin. 1–42 (2017)
16. Chen, H., Zhang, Y., Kalra, M.K., Lin, F., Chen, Y.: Recent advances in image reconstruction for PET. J. Nucl. Med. **60**(5), 661–682 (2019)
17. Mahmood, F., Kaiser, M.S.: A review of deep learning in medical imaging: Image acquisition to diagnosis. J. Healthc. Eng. (2018)
18. Greenspan, H., van Ginneken, B., Summers, R.M.: Guest editorial deep learning in medical imaging: Overview and future promise of an exciting new technique. IEEE Trans. Med. Imaging **35**(5), 1153–1159 (2016)
19. Goodfellow, I., Bengio, Y., Courville, A.: Deep Learning. MIT Press (2016)
20. Zhao, L., Feng, J., Gao, X., Zhuang, T., Hu, Y., Wu, F.: Artificial intelligence in medical imaging: technical aspects and clinical applications. Front. Med. **11**(1), 27–33 (2017)
21. Shen, D., Wu, G., Suk, H.I.: Deep learning in medical image analysis. Ann. Rev. Biomed. Eng. **19**, 221–248 (2017)
22. Hamet, P., Tremblay, J.: Artificial intelligence in medicine. Metabolism **69S**, S36–S40 (2018)
23. Shen, D., Wozniakowski, T., Zhou, X.S.: Deep learning for healthcare: a review on recent advances (2019). arXiv preprint arXiv:1906.08926
24. Marchesi, C.: Artificial Intelligence in Healthcare: A Primer. Springer (2020)
25. Rajkomar, A., Duan, Y., Gorodilov, D., Wei, Y., Sun, J., Peng, J., Shen, D., Erlich, H.R., Fine, J.P., Liu, H.: Scalable and Accurate Deep Learning for Risk Prediction of All-Cause Mortality in Electronic Health Records. Public Library of Science (2018)

Machine Learning Approach Versus AutoML to Predict the Bioactivity of a Therapeutic Target Related to Cancer

Abdellah Idrissi, Khawla Elansari, and Fatima Zahra El Houti

Abstract This study presents a machine-learning approach to develop potent EGFR inhibitors for breast cancer treatment, leveraging data from the ChEMBL database and employing chemoinformatics and RDKit for model generation. Despite achieving high accuracy, the predictive model's R^2 score suggests room for improvement. The research compares traditional machine learning models (e.g., SVR, Random Forest, XGBoost, CatBoost) against automated machine learning (AutoML) tools like Pycaret and H2O AutoML, focusing on efficiency and performance metrics (RMSE, MAE, R^2). Findings indicate that while the Random Forest model excels in traditional settings, AutoML offers significant time-saving advantages, albeit with slightly lower performance. This work underscores the potential of integrating AI into drug discovery, balancing between manual expertise in model building and the accelerated development process afforded by AutoML.

1 Introduction

Breast cancer remains one of the most prevalent and devastating forms of Cancer among women worldwide, posing significant health challenges and necessitating ongoing research into more effective treatments [1]. Among the myriad of targets for therapeutic intervention, the Epidermal Growth Factor Receptor (EGFR) has emerged as a critical player in the proliferation and survival of cancer cells. Inhibition of EGFR signaling pathways offers a promising avenue for developing targeted therapies for breast cancer [2]. However, discovering and optimizing potent EGFR inhibitors are complex and resource-intensive processes that require integrating diverse biochemical data and sophisticated analytical methods.

Recent advancements in machine learning (ML) and artificial intelligence (AI) have revolutionized many fields, including drug discovery and development. These

A. Idrissi (✉) · K. Elansari · F. Z. E. Houti
Artificial Intelligence & Data Science Group, IPSS Team, Computer Science Laboratory (LRI), Computer Science Department, Faculty of Science of Rabat, Mohammed V University in Rabat, Rabat, Morocco
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_28

technologies offer powerful tools for analyzing vast datasets, predicting molecular interactions, and optimizing drug candidates with unprecedented speed and accuracy [3]. Specifically, machine learning models can efficiently process and learn from chemical and biological data, facilitating the identification of potential compounds that can serve as effective EGFR inhibitors.

This study explores the potential of a machine learning-based approach to accelerate the discovery and development of potent EGFR inhibitors for treating breast cancer. By leveraging chemoinformatics techniques and comprehensive datasets from the ChEMBL database, we aim to design, synthesize, and evaluate compounds with high inhibitory activity against EGFR. Our approach utilizes RDKit to generate molecular fingerprint descriptors [4], forming the basis for developing EGFR target bioactivity predictive models.

Moreover, we investigate the application of automated machine learning (AutoML) [5] in optimizing the drug design process. AutoML represents an evolution in machine learning methodologies, automating complex model development tasks and enabling researchers to focus on the biological implications of their findings [6]. Through a comparative analysis of traditional machine learning models and AutoML solutions, this study highlights the efficiencies and potential improvements in predictive modeling that these technologies offer.

In doing so, this research contributes to the ongoing efforts to combat breast cancer by providing insights into the application of machine learning in drug discovery. By integrating AI with chemoinformatics and pharmacological research, we aim to streamline the development of EGFR inhibitors, paving the way for more effective and personalized cancer treatments.

2 Related Work

2.1 Existing Study

Artificial intelligence (AI) in drug discovery represents a significant advancement in pharmacology and computational biology. Leveraging chemoinformatics and machine learning (ML) techniques, researchers have developed models to predict the bioactivity of potential drug compounds against the epidermal growth factor receptor (EGFR), a key target in breast cancer therapy. The study utilizes datasets from the ChEMBL database, processed through RDKit, to generate molecular fingerprint descriptors, a method proven to enhance the accuracy of predictive models in drug discovery [3, 4, 6, 7].

However, despite achieving high accuracy, the R^2 scores indicate room for model improvement, highlighting a gap in the predictive capability of current models. This challenge underscores the need to refine machine-learning approaches in the pharmacological context [3] (Fig. 1).

algorithm	type	description of hyperparameter setting	R^2 train	R^2 test
Random Forest	ensemble learning	*n*_estimators = 100, random_state = 42	0.959	0.717
Linear Regression	linear model		0.693	0.568
Ridge Regression	linear model	alpha = 1.0	0.693	0.573
Lasso Regression	linear model	alpha = 1.0	0.057	0.056
Elastic Net	linear model	alpha = 1.0, l1_ratio = 0.5	0.058	0.057
K-NN Regression	instance-based	*n*_neighbors = 5	0.494	0.212
SVM Regression	support vector	kernel = "linear", *C* = 1.0	0.614	0.489
MLP Regression	neural network	hidden_layer_sizes = (100), activation = "relu"	0.922	0.596
XGBoost	boosting	*n*_estimators = 100, learning_rate = 0.1, max_depth = 3	0.898	0.704
LightGBM	boosting	*n*_estimators = 100, learning_rate = 0.1, max_depth = 3	0.797	0.681
CatBoost	boosting	*n*_estimators = 100, learning_rate = 0.1, max_depth = 3	0.619	0.569

Fig. 1 Machine learning algorithms used in the article [8]

2.2 Theoretical Framework

Building a machine learning model involves making decisions about data preprocessing, algorithm choice, and hyperparameter selection. AutoML (automated machine learning) automates the process, reducing the development costs associated with machine learning algorithms.

The foundation of AutoML is built upon various components and methodologies:

First, a configuration space defines "what" to automate. It encompasses data preprocessing, model selection, and the choice and optimization of algorithms, laying the groundwork for addressing AutoML challenges [9].

- Data Preparation: Essential for enhancing feature predictive power, data preparation includes hyperparameter tuning for dimensionality reduction and the selection and generation of features.
- Model Selection: AutoML facilitates the discovery and fine-tuning of model hyperparameters to ensure optimal performance, automating what is often a manually intensive process.
- Algorithm Optimization: Identifies the optimal types of algorithms and their hyperparameters, constituting the primary focus during the optimization phase.

Next, the controller defines "how" we solve an AutoML problem. It consists of an optimizer, which leverages the defined search space to generate viable configurations efficiently, and an evaluator responsible for assessing the effectiveness of these configurations. Despite being the most resource-intensive phase, recent advancements in AutoML have introduced evaluation techniques that expedite this process without compromising accuracy, thereby minimizing computational expenses [10] (Fig. 2) as presented in [11].

3 Methodology

The methodology used in this study includes several tools and techniques. The primary data source for this research study is the ChEMBL database, a biochemical database that gathers information on the bioactivity of compounds about drug

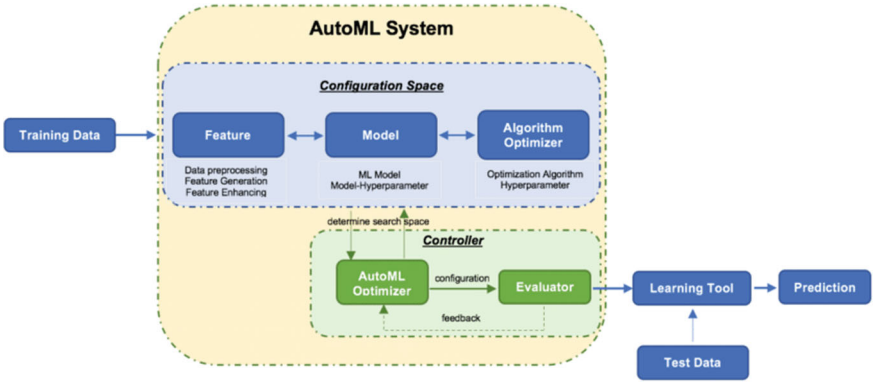


Fig. 2 AutoML system workflow [11]

targets to facilitate the discovery of new drugs. It provides a dataset on the compounds needed for our analysis.

To extract the molecular characteristics of chemical compounds, we used RDKit and PaDEL-Descriptor software. As part of the comparative study tween the traditional machine learning approach and AutoML, and given that the survey uses techniques to model and predict the bioactivity of the EGFR therapeutic target, we chose the SVR, Random Forest, XGBoost, and CatBoost models for the first approach. In contrast, for the second approach, AutoML, we used the Pycaret and H2O Automl tools.

We followed the same steps in both processes to implement both machine learning approaches. We used the Optuna library for model optimization, which enabled us to optimize model hyperparameters by running several trials with different combinations to improve model performance. Finally, we evaluated the models using the RMSE, MAE, and R^2 performance measures.

3.1 Data Acquisition

As mentioned, the data are extracted from the ChEMBL database, focusing on the EGFR target. The data come specifically from human samples. Information stored includes ChEMBL identifiers and standardized chemical structure representations of canonical_smiles molecules. We subsequently extracted Lipinski's molecular descriptors from the data. To characterize the compounds and use them as features in the model, we employed PADEL molecular fingerprints for their ability to concisely represent the characteristics of the molecules, facilitating analysis and modeling in our study.

The process of obtaining the final dataset took place in two stages: the calculation of Lipinski descriptors, followed by the calculation of molecular fingerprints, thus completing the creation of the dataset ready for model building.

3.2 Data Mining

We specifically performed data mining within the Lipinski descriptor dataset because of the relevance of these descriptors for target variable prediction and molecular fingerprint generation. This step, aimed at graphically visualizing the data, is crucial before constructing machine learning algorithms. Lipinski's descriptor data distributions were presented as histograms, while we visualized extreme values using boxplots, each representing an explanatory variable. This exploration then guided the preprocessing of the data to generate molecular fingerprints, forming the basis for predicting the bioactivity of therapeutic targets.

3.3 Data Preprocessing

The data preparation involved several steps, including managing missing data and data transformation. In addition, we performed feature selection using the SelectKBest technique with f_classif, based on the analysis of the variance test, for which we set up a loop that adjusts the k parameter to determine the optimal number of features. The aim is to identify a set of significant variables that will enable us to obtain the best model for minimizing the mean squared error on a test set. This approach helped us to select the features that contribute to optimizing the overall performance of the machine learning model.

Concerning AutoML, we explored the data preprocessing methods proposed by AutoML frameworks for feature selection on molecular fingerprint data to assess their relevance and effectiveness in the context of our project.

3.4 Model Implementation

As regards the learning and test bases, we have partitioned our data set into two distinct parts. The training sample, representing 80% of the data, and the test sample, covering the remaining 20%, are reserved for model evaluation.

We selected four models for the classical ML approach: Support Vector Regressor (SVR), Random Forest (RF), XGBoost, and CatBoost. Regarding the AutoML approach, we opted for the Pycaret and H2O AutoML frameworks, which can automatically select and train different models, offering suggestions for various models.

Table 1 Results of different ML models

Metric / Model	RMSE	MAE	R^2	Training time
Model 1–SVR	0.84	0.50	0.69	1 h 10 min 53 s
Model 2–RF	0.64	0.45	0.82	1 h 15 min 9 s
Model 3–XGboost	1.04	0.74	0.52	15 min 44 s
Model 4–Catboost	1.05	0.73	0.51	2 h 43 min 24 s

Interestingly, some of the models suggested by the AutoML frameworks were similar to those we had chosen in our classic ML approach.

Model performance evaluation in this study is based on three main metrics: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and coefficient of determination (R^2).

This identical approach to the two approaches guaranteed a comparison between the two, ensuring consistency in the development of the models and facilitating a precise evaluation of the performance of each method.

4 Results and Discussion

4.1 Classical ML Approach

The results from the classical machine learning approach provide a detailed overview of the performance of the selected models. The table below shows the results of the four models trained on the same data collection. To begin with, we observe that better results were obtained with model 2, "Random Forest", in particular for the lower RMSE measure and a higher R^2 compared to the other models, which equals 0.82. The fourth model has similar results to the third model, but it takes a very long time to train compared to the other models (Table 1).

4.2 Results of the AutoML Approach

The results of the AutoML approach reveal the outstanding performance of the best models identified by the Pycaret and H2O AutoML frameworks. This analysis shows the effectiveness of Automl, revealing the optimal model for each of the two tools. Regarding training time, both frameworks, PyCaret and H2O AutoML, offer low-code environments, contributing significantly to optimizing the time required to complete modeling tasks. Note that these tools facilitate the search and selection

Table 2 Results for the best AutoML models

Metric Model	RMSE	MAE	R^2	Training time
StackedEnsemble H2O	0.97	0.71	0.59	218,490 ms
RandomForest Pycaret	1.02	0.72	0.51	383,920 ms

of the most suitable models for our problem, offering a more efficient approach to solving modeling challenges.

The table below shows that H2O AutoML performed slightly better than PyCaret but was still significantly better in accuracy; this is because H2O AutoML supports two types of grid search, classical (or "Cartesian") grid search and random grid search, to optimize hyperparameters (Table 2).

4.3 Comparison of Results Between the Two Approaches, ML and AutoML

4.3.1 Performance

To assess performance, we compared the trained models of the two approaches. For the Automl approach and the PyCaret framework, the Random Forest model was identified as the best performer. As for the second framework, the best model obtained was a set of models called StackedEnsemble, with a coefficient of determination R^2 of 0.59. However, the Random Forest model of the classic ML approach stood out as the best-performing among the models evaluated. In contrast, the other models, such as XGBoost and CatBoost, performed reasonably similarly to the different models of the AutoML approach (Table 3).

The results show that the Random Forest model generated by the classic ML approach demonstrates a notable superiority over that generated by AutoML. The AutoML-generated models also showed inferior performance. This finding underlines the importance of manual expertise in the classical approach to model building, highlighting the ability of traditional approaches to outperform automatic ones in certain situations.

Table 3 Comparison of model results for the two approaches

Metrics	ML				AutoML		
	SVR	RF	XGboost	Catboost	St.E H2O	DRF H2O	RF Pycaret
RMSE	0.84	0.64	1.04	1.05	0.97	0.98	1.02
MAE	0.50	0.45	0.74	0.73	0.71	0.71	0.72
R^2	0.69	0.82	0.52	0.51	0.59	0.57	0.51

Table 4 Comparison of training times for the two approaches

	ML				AutoML		
	SVR	RF	XGB	Catboost	St.E H2O	DRF H2O	RF Pycaret
Training time (ms)	4,253,000	4,509,000	944,000	9,804,000	21,849	22,705	38,392

4.3.2 Training Time

The training time results show the advantages of the AutoML approach over the classical ML approach. For models in the classical ML approach, the training time is longer. This means that these models require higher computational resources and more training time.

In contrast, AutoML models show much shorter training times. This radical reduction in training time is one of the significant advantages of AutoML, as it enables faster iteration during model development (Table 4).

4.3.3 Configuration Complexity

Configuring machine-learning models is an essential step in the machine-learning process. It begins with selecting the most appropriate algorithms and their in-depth understanding, which requires considerable time. It also involves selecting features and adjusting hyperparameters to achieve better performance. In the classical approach, we used the Optuna framework to optimize hyperparameters, requiring in-depth knowledge of the algorithms and their fundamental parameters. The variety of hyperparameters in each model introduced an additional level of complexity.

Moreover, many lines of code are needed to configure complex models, involving a series of trial-and-error, consuming much time and effort. The use of AutoML has considerably simplified this phase. The tools used have automated finding and optimizing hyperparameters, thus reducing configuration and code complexity. This automation saved precious time and resulted in models with less effort.

5 Conclusion and Prospects

In conclusion, this comparative study between the traditional machine learning approach and AutoML has enabled us to meet our research objective of investigating both methods and identifying the advantages and limitations of each approach.

The contribution of this research can be seen in the improvement of the results of the reference article thanks to a specific selection of features and the optimization of hyperparameters with the Optuna library. This combination considerably improved

the performance of the Random Forest model, achieving an R^2 score of 0.82, an increase of 0.11 over the reference model.

Our study underlines the simplicity of AutoML in the algorithm selection process and the fact that this approach should be seen as a support tool rather than a total alternative. Our research identified certain limitations of the AutoML approach, such as primary data preprocessing and the need for more transparency associated with automatic algorithm selection.

An essential prospect lies in improving data preprocessing, given that the current choice of techniques is crucial. Concentrating on this critical phase could enhance data quality, significantly impacting overall model performance. In parallel, optimizing hyperparameters in AutoML represents another area to be explored. Concentrating on these two aspects would make it possible to obtain more accurate and adapted models, thus reinforcing the overall efficiency of the AutoML approach. In this way, we can also adapt the approaches mentioned in [12–26] to have more interesting results.

References

1. Siegel, R.L., Miller, K.D., Jemal, A.: Cancer statistics, 2020. CA: Cancer J. Clin. **70**(1), 7–30 (2020). https://doi.org/10.3322/caac.21590
2. Normanno, N., De Luca, A., Bianco, C., Strizzi, L., Mancino, M., Maiello, M.R., Carotenuto, A., De Feo, G., Caponigro, F., Salomon, D.S.: Epidermal growth factor receptor (EGFR) signaling in Cancer. Gene **366**(1), 2–16 (2006). https://doi.org/10.1016/j.gene.2005.10.018
3. Cherkasov, A., Muratov, E.N., Fourches, D., Varnek, A., Baskin, I.I., Cronin, M., Dearden, J., Gramatica, P., Martin, Y.C., Todeschini, R., Consonni, V., Kuz'min, V.E., Cramer, R., Benigni, R., Yang, C., Rathman, J., Terfloth, L., Gasteiger, J., Richard, A., Tropsha, A.: QSAR modeling: where have you been? Where are you going to? J. Med. Chem. **57**(12), 4977–5010 (2014). https://doi.org/10.1021/jm4004285
4. Landrum, G.: RDKit: Open-source cheminformatics (2016). Retrieved from https://www.rdkit.org
5. He, X., Zhao, K., Chu, X.: AutoML: A survey of the state-of-the-art. Knowl.-Based Syst. **212**, 106622 (2021)
6. Olson, R.S., Bartley, N., Urbanowicz, R.J., Moore, J.H.: Evaluation of a tree-based pipeline optimization tool for automating data science. In: Proceedings of the Genetic and Evolutionary Computation Conference 2016, pp. 485–492. ACM (2016). https://doi.org/10.1145/2908812.2908918
7. Gaulton, A., Hersey, A., Nowotka, M., Bento, A.P., Chambers, J., Mendez, D., et al.: The ChEMBL database in 2017. Nucl. Acids Res. **45**(D1), D945–D954 (2017)
8. Probst, P., Wright, M.N., Boulesteix, A.L.: Hyperparameters and tuning strategies for random forest. Wiley Interdiscip. Rev.: Data Min. Knowl. Discov. **9**(3), e1301 (2019)
9. Hutter, F., Kotthoff, L., Vanschoren, J.: Automated Machine Learning: Methods, Systems. Springer, Challenges (2019)
10. Feurer, M., Hutter, F.: Hyperparameter optimization. In: Automated Machine Learning, pp. 3–33. Springer (2019)
11. Krishna, V., Shrestha, Y.R., von Krogh, G.: Integrating advanced machine learning in information systems research: what can automated machine learning and transfer learning offer? (2021). http://doi.org/10.2139/ssrn.3855652

12. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv preprint arXiv:1307.5910
13. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
14. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
15. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
16. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF, 107–116 (2006)
17. El Handri, K., Idrissi, A.: Parallelization of Top_k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
18. El handri, K., Idrissi, A.: Comparative study of Top_k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
19. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
20. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
21. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
22. Abourezq, M., Idrissi, A.: A cloud services research and selection system. In: IEEE ICMCS (2014)
23. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
24. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
25. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
26. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless technologies (2016)

Integrating Artificial Intelligence with Information Systems in Healthcare Supply Chain Management

Sabrina Guetibi

Abstract This chapter investigates the transformative potential of integrating Artificial Intelligence and Information Systems in Healthcare Supply Chain Management. Drawing insights from recent studies delve into specific AI applications such as predictive analytics, demand forecasting, and decision support systems within healthcare SCM. Some of the recent studies investigate on how AI-driven technologies optimize inventory management, anticipate patient needs, and improve decision-making processes, ultimately enhancing operational efficiency and patient care outcomes. However, challenges such as data interoperability, regulatory compliance, and workforce readiness must be addressed to fully utilize the capabilities of AI in healthcare SCM. By re-thinking data strategy, ensuring regulatory compliance, and investing in workforce prior and post-training, healthcare organizations can address these obstacles and achieve the complete advantages of AI integration in SCM.

Keywords Healthcare supply chain management · Artificial intelligence · Information systems · Operational efficiency · Patient care

1 Introduction

Artificial Intelligence (AI) has transformed Healthcare Supply Chain Management (SCM) with its diverse applications, including predictive analytics, demand forecasting, and decision support systems. These technologies optimize inventory management, anticipate patient needs, and enhance decision-making processes within healthcare organizations [7]. Recent studies illustrate AI's impact, from predictive maintenance of medical equipment to dynamic inventory management

S. Guetibi
SIST British University, Associate College of Cardiff Metropolitan University, Tangier, Morocco

S. Guetibi (✉)
Artificial Intelligence & Data Science Group, IPSS Team, Faculty of Science of Rabat, Mohammed V University, Rabat, Morocco
e-mail: sabrina.guetibi@gmail.com; sabrina.guetibi@sist.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_29

of pharmaceuticals, highlighting its tangible benefits in improving efficiency and patient care within healthcare SCM.

The actual state of Healthcare SCM faces challenges related to efficiency and patient care outcomes. Integrating AI with Information Systems (IS) offers a promising solution to address these challenges. This chapter explores the reframing potential of AI integration in healthcare SCM, drawing insights from recent studies.

The integration of AI with IS in Healthcare SCM is to revolutionise the industry, by offering significant benefits such as improved operational efficiency and enhanced patient care outcomes. However, this integration presents challenges that need to be addressed for AI to reach its full potential in healthcare SCM.

This chapter explores how AI, when integrated with IS, revolutionizes SCM processes in healthcare, enabling proactive decision-making and fostering a more efficient and patient-centered supply chain ecosystem. By examining specific AI applications and real-world examples in healthcare settings, this chapter illuminates the immense potential of AI in optimizing SCM operations and improving patient outcomes while addressing challenges associated with its implementation.

This research objective is to explore how the integration of AI and IS in Healthcare SCM can enhance operational efficiency and patient care outcomes, delving into the present challenges related to data interoperability, regulatory compliance, and workforce readiness.

2 Literature Review on AI Integration in Healthcare

The integration of AI within IS presents a significant opportunity for transformation within Healthcare SCM [12].

AI has a multitude of applications in healthcare SCM, offering tools like predictive analytics, demand forecasting, and decision support systems. These technologies can optimize inventory management, anticipate patient needs, and enhance decision-making processes, ultimately improving operational efficiency and patient care outcomes [7]. These papers, Cannavale et al. [5], Almehmadi [4], Kumar et al. [13] Aldoseri et al. [2], Ji et al. [11], Raimo et al. [15], Uysal [18], Ali and Kannan [3]…, illustrate how healthcare organizations use AI-driven technologies to anticipate patient needs, manage inventory levels, and make informed decisions throughout supply chain logistics. Examples range from predictive maintenance of medical equipment to dynamic inventory management of pharmaceuticals, highlighting the tangible benefits of AI in enhancing efficiency and patient care within healthcare SCM processes. Cannavale et al. [5] emphasize the importance of AI in fostering adoption of innovation in healthcare networks involving multiple organizations. They assert that AI serves as a transformative tool, enhancing buyer-supplier relationships and driving performance improvements. By enabling operators to access crucial supplier information like pricing, stock availability, and delivery status, AI helps mitigate information asymmetry by providing operators with crucial supplier information, such as pricing and stock availability, thereby fostering business relationships

and facilitating vertical alliances along the value chain between buyers and suppliers. This promotes transparency and provides real-time information, empowering healthcare actors to analyze the rationale behind their operational decisions, here comes joining the power of real time information management into enterprise collaboration systems to enhance the efficiency of SCM operations [5]. Furthermore, recent research highlights the application of AI in health informatics education in Arab countries, particularly in Saudi Arabia [4], the research examines health informatics (HI) educational offerings in various Arab nations, observing that these programs are primarily provided by public institutions affiliated with medical faculties. In Saudi Arabia, the College of Computer and Information Systems at Umm Al-Qura University in Makkah is actively involved in offering HI programs, with a significant focus on AI. This reflects the rapid advancements in information technology and the growing importance of information management roles in healthcare. Demonstrating how AI is integrated into education to prepare students/future professionals for the evolving healthcare landscape in Arab countries. In the other hand, Kumar et al. [13] investigate the use of AI in healthcare supply chain (HSC) management, with a specific focus on identifying the critical success factors (CSFs) for AI adoption in emerging economies. The authors employed the Rough SWARA method to prioritize these CSFs and found that technological factors had the most significant impact on AI adoption in the HSC, with institutional or environmental factors, human factors, and organizational dimensions following suit. The study underscores AI as a game-changer in healthcare supply chain management, offering benefits such as improved quality of care, better inventory management, scams reduction, and enhanced security and traceability.

The integration of AI with existing IS in healthcare is essential for effectively leveraging AI technologies. This integration requires a rethinking of data strategy and integration to ensure seamless data exchange and interoperability. Practical examples demonstrate how this integration can revolutionize patient care and SCM efficiency. Drawing insights from recent studies, exploring the collaborative potential of AI and information systems in healthcare, Aldoseri et al. [2] emphasised on the critical role of data strategy and integration in leveraging AI technologies effectively within healthcare settings.

Practical examples illustrate how seamless data exchange and integration facilitate AI-driven insights, improving decision-making processes and patient outcomes. By rethinking data strategy and integration, healthcare organizations have the opportunity to fully leverage AI to transform patient care and enhance the efficiency of SCM. Ji et al. [11] discuss the importance of rigorous evaluations of clinical decision support systems (CDSSs) in determining their success and diffusion. Their study synthesizes 39 measures from 45 systematic reviews published between 2009 and 2020, identifying essential characteristics critical to the effectiveness of information systems, as outlined in the DeLone and McLean IS Success Model. The findings highlight the need for a comprehensive evaluation framework for CDSSs to ensure their development, utilization, and research are effective [11]. Additionally, the integration of AI with information systems is crucial for driving digital transformation in the healthcare industry, as highlighted by Raimo et al. [15]. The authors highlight

the growing significance of digital healthcare technologies, particularly in response to the COVID-19 pandemic, noting their potential to mitigate healthcare disparities, enhance healthcare standards, and boost public well-being. They focus their research on the degree of digital transformation occurring in hospitals in Italy, pinpointing factors such as hospital size, age, teaching status, and complexity that influence the uptake and integration of digital healthcare solutions. Moreover, Uysal [18] emphasizes the importance of adopting the guidelines of Industrial Information Integration Engineering (IIIE) for designing, developing, Implementing healthcare information systems (HEIS) enabled with Machine Learning (ML) technology. The study discusses the challenges and risks associated with ML enabled systems, highlighting the need for a structured approach to integration, especially in the context of healthcare. Following the guidelines of Action Research, Design Science Research, and IIIE, the study proposes an integrated architecture for ML-enabled HEIS in a university hospital, aiming to address the complexity and integration issues of these systems. In their paper, Ali and Kannan [3] discuss the use of machine learning in IS and SCM within the context of healthcare operations. The authors highlight the importance of machine learning in enhancing decision-making processes and improving efficiency in healthcare systems. For instance, machine learning has been utilized to enhance various aspects of healthcare operations, such as optimizing patient waiting times through simulation and Markov models [6, 20], enhancing surgical processes [19], controlling patient flow in emergency departments [1], and streamlining outpatient appointment scheduling [17]. ML techniques have been utilized to address challenges stemming from the COVID-19 pandemic, including supplier selection [14], the design and planning of testing facilities [8], and the development of public health intervention frameworks [9], the authors recommend that future studies concentrate on constructing integrated models that examine various factors influencing patient waiting times, such as overall service time, equipment failures, and staff turnover. Additionally, the authors propose exploring the use of Industry 4.0 technologies, including the Internet of Things (IoT) and big data analytics in healthcare to enhance decision-making processes. This includes exploring their potential in contact tracing and outbreak identification during pandemics like COVID-19.

Hackl et al. [10] discuss the application of ML in Clinical Information Systems (CIS) and its potential link to Supply Chain Management (SCM) and Information Systems (IS). The authors highlight the role of machine learning in enhancing healthcare delivery and patient outcomes within CIS. They suggest that the integration of machine learning in CIS could lead to more efficient and patient-oriented healthcare systems. While the paper does not directly discuss the integration of machine learning in SCM or IS, it implies that advancements in machine learning within CIS could have broader implications for IS, including SCM. By improving data analysis and decision-making processes in healthcare, machine learning could potentially enhance supply chain operations and overall information management in healthcare organizations, which are key components of IS.

Sharma et al. [16] examined the role of AI in SCM and delineated its various applications. They highlighted that AI technologies, such as ML and natural language processing, have significant potential in SCM, and robotics, are increasingly being

used in SCM to improve operational efficiency, enhance decision-making processes, and optimize supply chain performance. The authors emphasized that AI can play a transformative role in SCM by enabling predictive analytics, automation of routine tasks, and optimization of complex processes. They discussed various applications of AI in SCM, such as demand forecasting, inventory management, logistics optimization, and selection of suppliers. Sharma et al. [16] also tackled the challenges and opportunities linked to adopting AI in SCM. They underscored the significance of data quality, privacy, and security in AI-driven SCM systems and emphasized the need for organizations to develop the necessary capabilities to effectively leverage AI technologies.

Overall, the authors concluded that AI has the potential to revolutionize SCM practices and drive significant improvements in efficiency, cost-effectiveness, and customer satisfaction. They recommended that organizations commit to investing in AI technologies and develop strategies to integrate AI into their SCM processes to stay competitive in the rapidly evolving business landscape.

3 Discussion

The integration of AI with IS in Healthcare SCM offers significant benefits, including improved operational efficiency and enhanced patient care outcomes. However, managing healthcare supply chains through AI presents several challenges that need to be addressed to fully realize its potential in healthcare SCM.

In the context of ML in healthcare operations and supply chain management, challenges related to AI include:

Data quality and availability: are critical factors for ML models, requiring substantial amounts of high-quality data to train effectively. Ensuring data quality and availability, especially in healthcare where data may be sensitive or fragmented across different systems, can be a significant challenge [16].

Model Interpretability and Transparency: ML models, particularly deep learning models, are often considered "black boxes" because they lack transparency in how they arrive at their decisions. In healthcare, where decisions can have critical implications for patient care, understanding and interpreting model decisions are crucial [16].

Integration with existing healthcare systems and workflows can be challenging when incorporating machine learning models. Ensuring seamless integration and interoperability with electronic health records (EHRs) and other systems is a complex task [3].

Ethical and Legal Considerations, ML models in healthcare raise ethical questions around patient privacy, consent, and bias. Ensuring that ML applications adhere to ethical standards and comply with legal regulations is essential [10].

Cost and Resource Constraints: Implementing machine learning in healthcare can be costly, requiring investments in technology, infrastructure, and skilled personnel.

Limited resources and budget constraints can hinder the implementation of machine learning in healthcare SCM [16].

Addressing these challenges is crucial for the successful integration of ML in healthcare operations and SCM, enabling organizations to leverage AI technologies effectively to improve patient care outcomes and operational efficiency. One key gap is the lack of standardized data formats and interoperability standards across different healthcare systems [2]. Healthcare organizations often use disparate systems that store data in different formats, making it challenging to integrate AI technologies seamlessly. To address this gap, healthcare organizations must develop and adopt standardized data formats and interoperability standards to ensure that AI technologies can effectively access and analyze data from diverse sources.

Additionally, regulatory compliance and data privacy concerns pose significant challenges to the integration of AI in healthcare SCM [13]. Healthcare organizations must comply with strict regulations regarding data privacy and security, which can limit the implementation of AI technologies. To overcome these challenges, healthcare organizations must develop robust data governance frameworks that ensure compliance with regulations while enabling the effective use of AI technologies.

Another critical gap is the lack of workforce readiness and training to effectively utilize AI technologies in healthcare SCM. Many healthcare professionals lack the necessary skills and knowledge to leverage AI-driven insights for decision-making and patient care [12]. To address this gap, healthcare organizations must invest in workforce training programs that educate healthcare professionals on the use of AI technologies and their impact on healthcare SCM. As it was highlighted by Almehmadi [4] and Kumar et al. [13] findings, both prior and post education of professional on the importance of AI and how it is transformative is primordial to the well establishment of this technology into Healthcare institution.

To bridge these gaps, healthcare organizations must focus on developing comprehensive data strategies that prioritize data quality, security, and interoperability. By investing in workforce training and regulatory compliance, healthcare organizations can leverage AI to revolutionize healthcare supply chain management and improve patient care outcomes.

4 Conclusion

The integration of AI with IS in healthcare SCM marks a significant advancement in healthcare delivery. By utilizing AI-driven technologies, healthcare organizations can enhance operational efficiency, resource utilization, and patient outcomes. However, several gaps and challenges, such as standardized data formats, regulatory compliance, and workforce readiness, must be addressed to fully realize the advantages of AI in healthcare.

In the realm of machine learning applications in healthcare operations and supply chain management, challenges such as ensuring data quality and availability, enhancing model interpretability and transparency, integrating with existing

systems, addressing ethical and legal considerations, and managing cost and resource constraints are prominent. Machine learning models necessitate high-quality data for effective training, and ensuring data quality and availability can be complex, especially in healthcare settings where data may be sensitive or dispersed across various systems.

Furthermore, machine learning models, particularly deep learning models, are often viewed as opaque "black boxes," lacking transparency in how decisions are made.This lack of clarity raises concerns within the healthcare sector, where understanding and interpreting model decisions are critical for patient care. Integrating machine learning models into current healthcare systems and workflows requires intricate integration and interoperability with electronic health records and other systems.

Additionally, ethical and legal issues concerning patient privacy, consent, and bias are paramount in machine learning applications in healthcare. Ensuring compliance with regulations while upholding ethical standards is essential for successful implementation. Finally, implementing machine learning in healthcare can be costly, necessitating investments in technology, infrastructure, and skilled personnel.

Addressing these challenges will enable healthcare organizations to harness the full potential of AI, transforming healthcare SCM and enhancing patient care outcomes. Investing in data strategies, workforce training, and regulatory compliance will be pivotal for healthcare organizations to overcome these challenges and revolutionize healthcare delivery globally.

References

1. Abo-Hamad, W., Arisha, A.: Simulation-based framework to improve patient experience in an emergency department. Eur. J. Oper. Res. **224**(1), 154–166 (2013)
2. Aldoseri, A., Al-Khalifa, K.N., Hamouda, A.M.: Re-thinking data strategy and integration for artificial intelligence: concepts, opportunities, and challenges. Appl. Sci. **13**, 7082 (2023). https://doi.org/10.3390/app13127082
3. Ali, I., Kannan, D.: Mapping research on healthcare operations and supply chain management: a topic modelling-based literature review. Ann. Oper. Res. **315**(1), 29–55 (2022). https://doi.org/10.1007/s10479-022-04596-5
4. Almehmadi, F.M.: Health information science and technology education: an analysis of health informatics undergraduate and postgraduate programs in Arab countries. Heliyon **9**, e19279 (2023). https://doi.org/10.1016/j.heliyon.2023.e19279
5. Cannavale, C., Tammaro, A.E., Leone, D., Schiavone, F.: Innovation adoption in inter-organizational healthcare networks—the role of artificial intelligence. Eur. J. Innov. Manag. **25**(6), 758–774 (2022). https://doi.org/10.1108/EJIM-08-2021-0378
6. Chae, B.K.: A general framework for studying the evolution of the digital innovation ecosystem: the case of big data. Int. J. Inf. Manag. **45**(2), 83–94 (2019)
7. Chan, C., Petrikat, D.: Strategic applications of artificial intelligence in healthcare and medicine. J. Med. Health Stud. **4**, 58–68 (2023). https://doi.org/10.32996/jmhs.2023.4.3.8
8. Fan, Z., Xie, X.: A distributionally robust optimisation for COVID-19 testing facility territory design and capacity planning. Int. J. Prod. Res., 1–24 (2022)
9. Ghaderi, M.: Public health interventions in the face of pandemics: network structure, social distancing, and heterogeneity. Eur. J. Oper. Res. **298**(3), 1016–1031 (2022)

10. Hackl, W.O., Neururer, S.B., Pfeifer, B.: IMIA Yearbook section on clinical information systems. Transforming clinical information systems: empowering healthcare through telemedicine, data science, and artificial intelligence applications. IMIA Yearb. Med. Inform., 127–137 (2023)
11. Ji, M., Yu, G., Xi, H., Xu, T., Qin, Y.: Measures of success of computerized clinical decision support systems: an overview of systematic reviews. Health Policy Technol. **10**, 196–208 (2021). https://doi.org/10.1016/j.hlpt.2020.11.001
12. Kersten, W., (Ed.), Blecker, T. (Ed.), Ringle, C.M., (Ed.): Artificial intelligence and digital transformation in supply chain management: innovative approaches for supply chains. In: Proceedings of the Hamburg International Conference of Logistics (HICL), No. 27, ISBN 978-3-7502-4947-9, epubli GmbH, Berlin (2019). https://doi.org/10.15480/882.2460
13. Kumar, A., Mani, V., Jain, V., Gupta, H., Venkatesh, V.G.: Managing healthcare supply chain through artificial intelligence (AI): a study of critical success factors. Comput. Ind. Eng. **175**, 108815 (2023). https://doi.org/10.1016/j.cie.2022.108815
14. Pamucar, D., Torkayesh, A.E., Biswas, S.: Supplier selection in healthcare supply chain management during the COVID-19 pandemic: a novel fuzzy rough decision-making approach. Ann. Oper. Res., 1–43 (2022)
15. Raimo, N., De Turi, I., Albergo, F., Vitolla, F.: The drivers of the digital transformation in the healthcare industry: an empirical analysis in Italian hospitals. Technovation **121**, 102558 (2023). https://doi.org/10.1016/j.technovation.2022.102558
16. Sharma, R., Shishodia, A., Gunasekaran, A., Min, H., Munim, Z.H.: The role of artificial intelligence in supply chain management: mapping the territory. Int. J. Prod. Res. **60**(24), 7527–7550 (2022). https://doi.org/10.1080/00207543.2022.2029611
17. Shehadeh, K.S., Cohn, A.E.M., Epelman, M.A.: Analysis of models for the stochastic outpatient procedure scheduling problem. Eur. J. Oper. Res. **279**(3), 721–731 (2019)
18. Uysal, M.P.: Machine learning-enabled healthcare information systems in view of industrial information integration engineering. J. Ind. Inf. Integr. **30**, 100382 (2022). https://doi.org/10.1016/j.jii.2022.100382
19. VanBerkel, P.T., Blake, J.T.: A comprehensive simulation for wait time reduction and capacity planning applied in general surgery. Health Care Manag. Sci. **10**(4), 373–385 (2007)
20. Vissers, J.M., Adan, I.J., Dellaert, N.P.: Developing a platform for comparison of hospital admission systems: an illustration. Eur. J. Oper. Res. **180**(3), 1290–1301 (2007)

Artificial Intelligence in Transport, IoT and Security

Comparative Analysis of Simultaneous Localization and Mapping Algorithms for Enhanced Autonomous Navigation

Slama Hammia, Anas Hatim, Abdelilah Haijoub, and Ahmed El Oualkadi

Abstract This paper presents a comprehensive analysis and comparative study of the methods and approaches employed in Simultaneous Localization and Mapping (SLAM). The goal is to shed light on the advantages, disadvantages, and trade-offs of various SLAM approaches. Numerous topics are covered in the study, such as different types of sensors, data processing algorithms, feature extraction techniques, and mapping frameworks. Examining their performance in terms of accuracy, resilience, computational efficiency, and adaptability for various settings and applications is given particular priority. This work intends to help researchers and practitioners in choosing the most suitable strategies for their particular SLAM requirements by examining the benefits and drawbacks of each strategy. Additionally, it identifies important issues and potential avenues for future study to promote SLAM technological developments. The thorough analysis and comparison done here help to clarify the current status of SLAM technology and make it easier to create better localization and mapping solutions.

Keywords SLAM · EKF-SLAM · Autonomous mobile robot · Extended Kalman filter · Localization and mapping · Implementation

S. Hammia (✉) · A. Hatim
TIM Team, ENSA-Marrakech, Caddi Ayyad University, Marrakech, Morocco
e-mail: hammia.slama@gmail.com

A. Hatim
e-mail: a.hatim@uca.ma

A. Haijoub
ESL, ENSA-Kenitra, Ibn Tofail University, Kenitra, Morocco
e-mail: haijoub.abdel@gmail.com

A. E. Oualkadi
ISI, ENSA-Tétouan, Abdelmalek Essaadi University, Tangier, Morocco
e-mail: aeloualkadi@uae.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_30

1 Introduction

In robotics and computer vision issues, simultaneous localization and mapping (SLAM) aims to allow autonomous systems to navigate and map their environments in real-time. Numerous approaches and strategies have been put out in the literature to solve the difficulties of SLAM due to the growing requirement for precise and reliable localization and mapping capabilities (Fig. 1). To help academics and practitioners choose the best approaches for their applications, this study gives a thorough analysis of the SLAM techniques now in use. This analysis offers insightful information on cutting-edge solutions. Recent academic studies that explored many areas of sensor types, data processing algorithms, feature extraction approaches, mapping frameworks, and sensor fusion techniques greatly contributed to the improvements in SLAM.

Because various sensor modalities gather diverse information about the environment, choosing the right sensors is essential in SLAM. Rich visual data is provided by visual sensors like cameras, which may be used for feature extraction and tracking. Range sensors, such as depth cameras or LiDAR (Light Detection and Ranging), provide precise distance measurements that allow for precise geometric mapping [1]. Designing efficient SLAM systems requires an understanding of the advantages and disadvantages of various sensor types.

To accurately estimate the robot's position and produce consistent maps, data processing methods are crucial in SLAM. Recursive estimating approaches are used in filter-based methods like the Extended Kalman Filter and Particle Filters to incrementally update the robot's state estimate [2]. The formulation of SLAM as an optimization problem in optimization-based techniques, such as Graph SLAM and bundle adjustment, allows for a cooperative estimate of the robot's trajectory

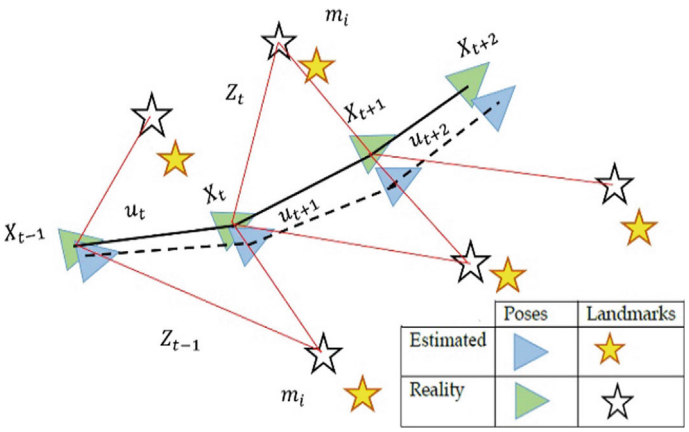


Fig. 1 Notations and illustration of the SLAM problem

and the underlying map [3]. It is possible to gain knowledge about these algorithms' suitability for use in various SLAM settings by comparing and assessing their performance.

The detection and tracking of distinctive visual or geometric elements over several sensor frames is made possible by feature extraction, a key component of SLAM [4]. Visual SLAM has made extensive use of key-point detection and matching algorithms to reliably track features, such as the Scale-Invariant Feature Transform (SIFT) and the Speeded-Up Robust Features (SURF) [5]. Techniques like point cloud registration and surface feature extraction are used in range-based SLAM to align subsequent scans and extract useful geometric data [6]. The relative advantages and disadvantages of various feature extraction techniques in distinct SLAM settings can be clarified by doing a thorough examination of those techniques.

The precision and effectiveness of the mapping process are influenced by mapping frameworks, which control how the world is represented and modeled in SLAM systems. By partitioning the environment into a grid and stating the occupancy probabilities for each cell, occupancy grid mapping offers a discrete representation of the environment. Truncated Signed Distance Fields (TSDF) and OctoMaps are two volumetric approaches that describe the environment as a continuous occupancy field, allowing for comprehensive 3D reconstruction [7]. The applicability of various mapping frameworks for various applications and contexts may be determined by analyzing and contrasting them.

The robustness and accuracy of SLAM systems are improved by merging data from several sensors using sensor fusion techniques Sensor fusion techniques can make use of each modality's advantages by merging data from complementing sensors, such as visual and range sensors [8]. Accurately estimating the robot's status requires the use of probabilistic frameworks, such as Kalman filter, Extended Kalman filter, or particle filters, which are frequently used to combine measurements [9] (Fig. 2). A thorough comparison of sensor fusion approaches can provide insightful information on their performance and applicability for various SLAM scenarios.

This study offers a thorough examination of the SLAM approaches currently in use. It covers a broad spectrum of topics, including sensor choice, data processing algorithms, feature extraction approaches, mapping frameworks, and sensor fusion methodologies. The insights offered in this study can help practitioners and researchers better grasp the benefits and drawbacks of various strategies, facilitating the choice of suitable techniques for certain SLAM applications.

2 Simultaneous Localization and Mapping (SLAM)

A crucial issue in the fields of robotics and computer vision is simultaneous localization and mapping (SLAM), which aims to allow autonomous systems to navigate and create maps of their environments concurrently. An in-depth discussion of SLAM's essential elements, difficulties, and various methodologies used in the literature are

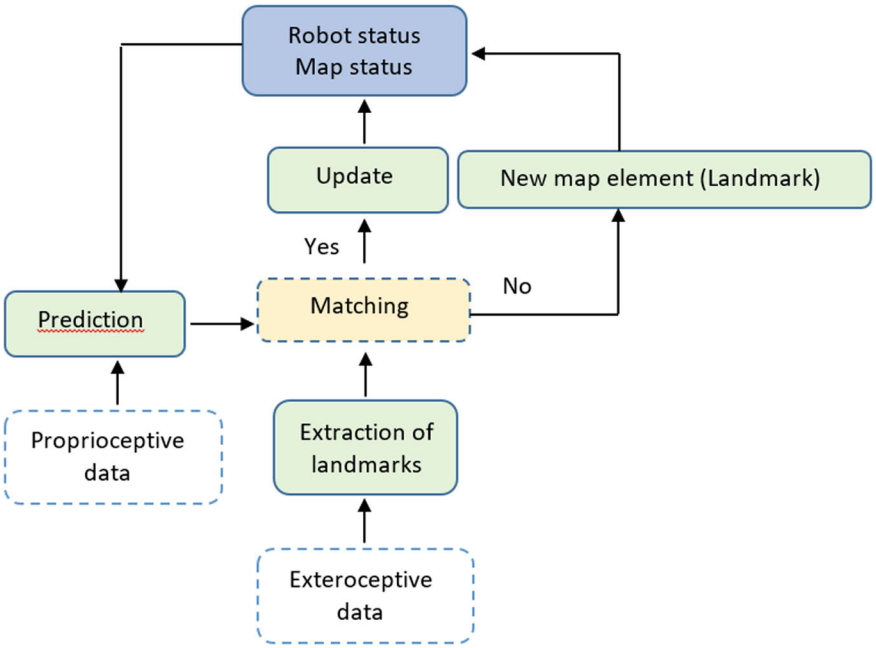


Fig. 2 Block diagram of the EKF-SLAM process

covered in this section. Understanding SLAM's complexities will help academics and practitioners better appreciate the advantages and limits of current methods, laying the groundwork for further advancements in this area.

Localization, which refers to determining the robot's position within the environment, and mapping, which focuses on creating a representation of the surrounding world, are the two main tasks involved in SLAM. For the robot to accurately navigate and plan a course, it needs to be able to localize itself with the mapped environment. The process of mapping, on the other hand, entails utilizing sensor data to capture and describe the structure of the environment. This produces an informative representation that may be used for a variety of activities, including scene comprehension and obstacle avoidance [10].

SLAM systems use sensor data and include various estimate and optimization methods to accomplish simultaneous localization and mapping. The surroundings and the mobility of the robot can be learned using sensor data collected from devices like cameras, LiDAR, or range sensors. The robot's position and a map of the area are then estimated using these measurements, which are subsequently processed and combined. A SLAM system's capabilities and limitations are significantly influenced by the sensors used, therefore picking the right sensor modality is essential to getting accurate and trustworthy results [11].

The foundation of SLAM systems is estimation algorithms, which allow the robot to deduce its pose and the structure of the surroundings from sensor readings. Probabilistic techniques can be used for estimation, such as the Kalman filter or particle filters, which simulate system uncertainty and update the belief over time. The most likely configuration of the robot's trajectory and map that minimizes the difference between anticipated and observed measurements is sought after by optimization-based approaches, such as Graph SLAM or bundle adjustment, which define SLAM as an optimization issue [12].

The data association issue, which entails connecting sensor readings to the matching features or landmarks in the environment, is one of the main difficulties in SLAM. For the robot's pose estimation to be updated accurately and a precise map to be created, this relationship is essential. The data association problem has been approached using a variety of ways, including feature-based methods that depend on distinguishing visual or geometric elements and direct methods that work directly on raw sensor data without explicitly relying on feature extraction [13].

Handling dynamic situations, where the presence of moving objects can dramatically impair the accuracy of localization and mapping, is another problem for SLAM. Dynamic objects introduce measurement uncertainty and may result in inconsistent map representation. Various approaches, including object identification and tracking, motion modeling, and map adaption algorithms, have been developed to manage dynamic settings [14].

Deep learning strategies have recently come to light as a potential direction for SLAM, utilizing the capability of neural networks to learn representations and predict outcomes directly from sensor data. In several fields, including visual SLAM, LiDAR-based SLAM, and sensor fusion, deep learning-based SLAM techniques have demonstrated promising results. Convolutional neural networks (CNNs) or recurrent neural networks (RNNs) are frequently used in these methods to extract data, predict the robot's position, or rebuild the environment [15].

Section 2 concludes by giving an overview of Simultaneous Localization and Mapping (SLAM), outlining its main elements, difficulties, and various strategies used in the literature. Researchers and practitioners may better appreciate the complexity of this discipline and look for possibilities for breakthroughs and strategies by knowing the SLAM foundations and developments.

3 Evolution of SLAM

Significant strides in the development of SLAM have been made thanks to algorithms and methods that address the difficulties of simultaneous localization and mapping. FastSLAM [16], a tailored approach that makes use of the idea of particle filters to solve the computational difficulty of SLAM, was one important development. FastSLAM achieves an effective and precise estimate of the robot's trajectory and the map by modeling the posterior distribution using a collection of particles.

The development of SLAM was significantly influenced by probabilistic robotics [17]. A thorough introduction of probabilistic methods for robotic perception and navigation, including SLAM, is provided in this essential work. Probabilistic SLAM algorithms, offer a theoretical framework and useful implementation recommendations, encouraging further developments in the area.

The introduction of Augmented Reality (AR) opened up new avenues for SLAM research. Real-time camera position tracking and 3D mapping in constrained AR workspaces were first introduced by Parallel Tracking and Mapping (PTAM) [18]. With the help of PTAM, SLAM's viability in interactive and dynamic settings was demonstrated, creating new opportunities for AR applications on portable devices.

By providing real-time single-camera-based localization and mapping, MonoSLAM [19] marks an important milestone in SLAM. To predict the camera posture and map structure using visual data, an extended Kalman filter is used. MonoSLAM paved the path for more monocular SLAM methods by showcasing the use of monocular vision in SLAM applications.

An adaptable and precise monocular SLAM system is ORB-SLAM [20]. It makes use of ORB attributes and descriptors for effective localization and mapping across a range of contexts. Bundle adjustment and loop closing techniques are used in ORB-SLAM to improve the predicted trajectory and map's accuracy and consistency.

The notion of large-scale direct monocular SLAM was first presented by LSD-SLAM [21]. LSD-SLAM accomplishes dense and accurate reconstruction of the environment without depending on explicit feature extraction by utilizing direct picture alignment and depth estimation. By radically improving monocular SLAM systems' perceptual skills, this method made it possible to create precise and accurate maps.

These foundational publications helped shape the development of SLAM by each offering particular ideas and methods for overcoming the difficulties of simultaneous localization and mapping. The discipline has made considerable strides in terms of accuracy, efficiency, and adaptability, from the effective particle filtering of FastSLAM to the probabilistic underpinnings put forth in Probabilistic Robotics. While MonoSLAM displayed the promise of monocular vision, PTAM showed the potential of SLAM in augmented reality applications. Monocular and direct monocular SLAM were both given robust and precise approaches by ORB-SLAM and LSD-SLAM, respectively. These developments have opened the door for more study and invention in the area, which has resulted in the creation of more complex SLAM techniques.

4 The Filtering Approach and the Smoothing Approach of SLAM

4.1 Filtering Approach

The recursive Bayesian filtering techniques, such as the extended Kalman filter (EKF) or the particle filter, are the foundation of the SLAM filtering approach, which is best shown by FastSLAM [22]. As fresh sensor readings become available, the filtering technique progressively updates an approximation of the robot's state, which includes its stance and map. For applications that call for online processing and little computing complexity, it offers real-time localization and mapping capabilities.

4.2 Smoothing Approach

In contrast, the smoothing strategy in SLAM concentrates on jointly estimating the whole trajectory of the robot and the underlying map by taking into account all available sensor measurements in a batch optimization framework. Smoothing-based SLAM systems frequently use nonlinear least squares solvers and factor graphs [23]. Particularly in settings with loop closures and long-term mapping applications, these approaches have the benefit of globally optimum estimations and enhanced accuracy.

4.3 Key Differences

The estimating procedures used by the filtering approach [22] and the smoothing approach [23] in SLAM show significant variations. As it analyzes data progressively, the filtering technique offers live estimates and is suitable for real-time applications. It could, however, have accumulated flaws and be inconsistent overall. The smoothing strategy, on the other hand, takes into account all available measurements at once, allowing for higher accuracy and global consistency at the expense of increased processing complexity and delayed estimates.

4.4 Incremental Smoothing and Mapping

A smoothing-based technique that tackles the processing difficulties by making use of the sparsity in the factor graph representation is incremental smoothing and mapping (iSAM2) [24]. Utilizing the Bayes tree data structure to effectively

absorb new measurements while preserving global consistency, iSAM2 progressively changes the solution. It fits applications that need both online processing and global consistency because it strikes a compromise between real-time performance and correctness.

4.5 Covariance Estimation and Optimization

Both filtering and smoothing methods depend heavily on covariance estimation since it expresses the degree of uncertainty in the predicted robot position and map. To increase the estimation's accuracy and resilience, methods like visual-inertial odometry (VIO) [25] integrate observations from inertial and visual sensors. Levenberg-Marquardt least squares solvers [26] and other optimization techniques are also used in the estimate phase to reduce the error between anticipated and observed data.

4.6 Comparative Evaluation

Understanding the relative advantages and disadvantages of the filtering and smoothing techniques used in SLAM is crucial. It is necessary to take into account variables like processing complexity, precision, scalability, and robustness to various circumstances. A paradigm for mobile mapping systems based on factor graph SLAM [27] sheds light on the benefits and drawbacks of the smoothing strategy.

This table gives a broad overview of the benefits and drawbacks, while unique implementation specifics and application circumstances may change it. When choosing a good SLAM approach for their unique needs, researchers and practitioners should take these variables into account and undertake a careful review (Table 1).

In summary, this section covered the SLAM filtering and smoothing approaches. While the smoothing technique [23] delivers globally optimum estimates and increased accuracy at the expense of increased computational complexity, the filtering approach [22] offers real-time localization and mapping capabilities with minimal computational complexity. A smoothing-based strategy that tackles the trade-off between real-time performance and global consistency is incremental smoothing and mapping [24]. Both approaches highlighted covariance estimation techniques [25] and optimization methods as essential elements. To evaluate the advantages and disadvantages of each strategy, a comparative assessment [27] is required.

Table 1 Comparison of different SLAM methods

SLAM Technique	Advantages	Disadvantages
Filtering-based	Instantaneously estimate	Errors in the linearization in nonlinear situations
(EKF, UKF)	Computing effectiveness Applications for real-time compatible	Limited precision in comparison to smoothing methods Adaptability to starting circumstances Difficulty in handling highly nonlinear scenarios
Smoothing-based	Higher accuracy	Higher computational demands
(RTS, GraphSLAM)	Includes all sensor measurements that are currently accessible Through backward passes, estimations are improved	Not suitable for real-time applications Potential scalability issues for large-scale mapping Longer processing time due to backward pass

5 Fundamental Implementation and Issue in SLAM

5.1 Visual Odometry

Visual odometry, which calculates the camera's ego motion by monitoring elements over successive frames, is a key element of SLAM. Fast semi-direct monocular visual odometry (SVO) [28] offers real-time performance and great accuracy by directly tracking sparse features frame by frame. Other approaches, such as ORB-SLAM [20], create the map and estimate camera motion by using feature extraction and matching algorithms. For the localization and mapping stages of SLAM, these visual odometry approaches give crucial data.

5.2 Parametrization Techniques

The accuracy and effectiveness of SLAM are significantly influenced by the parametrization method used to represent the environment's 3D structure. A popular approach, inverse depth parametrization [29], enables more reliable handling of uncertain depth data, particularly in monocular SLAM scenarios. It offers flexibility in adding new features and makes effective optimization strategies possible.

5.3 Direct SLAM

Because of the necessity of effectively processing and integrating raw sensory input, Direct SLAM implementation poses substantial technical challenges. The control of measurement errors, sensor calibration, and taking into consideration the environment's changing light and texture conditions are important concerns. Furthermore, ensuring dependable performance in real-world settings requires addressing the system's resilience to external perturbations such obstacles and viewpoint shifts [29].

5.4 Loop Closure

A key component of SLAM that deals with the issue of identifying and fixing accumulated drifts in the predicted trajectory is loop closure. Geometric and appearance-based approaches are used in robust loop-closing strategies, such as the one put out by Guclu et al. [30], to identify previously visited areas and create loop closures. Loop closure increases the SLAM system's accuracy and enables global map consistency.

5.5 Scalability and Real-Time Performance

Particularly in large-scale or dynamic situations, SLAM systems' scalability and real-time performance present considerable hurdles. A method called parallel tracking and mapping (PTAM) [18] makes use of distributed mapping and parallel processing to enable real-time operation even on devices with limited resources. Due to the precision and computational efficiency trade-off that PTAM offers, it is ideal for AR applications and tiny workspaces.

In the end, this section covered the essential SLAM implementation concerns and challenges as well as a comparison between different SLAM implementation methods and techniques in terms of resource consumption, which provides an overview of the complexity of each implementation. SVO [28] and ORB-SLAM [20] are two visual odometry methods that are crucial for calculating camera motion and building the map. Robust treatment of ambiguous depth data is made possible by parametrization approaches, such as inverse depth parametrization. Direct SLAM techniques, such as LSD-SLAM [21], act on pixel intensities directly and provide high-fidelity mapping. Loop closure methods, like those suggested by Guclu et al. [30], take care of accumulated drifts and guarantee map global consistency. Scalability and real-time performance are addressed by techniques like PTAM [21], which make optimal use of parallel processing. The effectiveness and applicability of SLAM systems are substantially impacted by these implementation issues and difficulties. A comparison of hardware resource usage between our design of the EKF-SLAM in

Table 2 Hardware resource consumption in different FPGA implementations of SLAM

Work	LUTs	Reduction in LUTs (%)	DSP	Memory blocks	Embedded multipliers	Registers	Reduction in registers (%)	Freq MHz
[31] 2022	18,577	–	0	0	212	2106	–	114
[32] 2021	22,174	16,22	43	3,272,051	–	–	–	717
[33] 2015	27,889	33,39	175	–	–	–	–	49,91
[34] 2020	28,433	34,66	87	3,970	174	77,650	97,29	125,42
[35] 2018	39,600	53,09	–	–	252	39,600	94,68	–
[36] 2020	418,080	95,56	2,520	–	–	–	–	200

[31] and other current techniques utilized in [32–35], and [36] is shown in Table 2. Hardware resources were significantly reduced in [31]. Both the number of registers and the number of LUT logic elements decreased, falling to 95.56% and 94.68%, respectively. The comparison demonstrates unequivocally that the implementation in [31] uses a very small amount of resources in contrast to other approaches.

6 Conclusion

In this paper, we conducted a thorough analysis and comparison of several SLAM techniques, with a focus on filtering and smoothing techniques. The goal of our investigation was to assess the advantages and disadvantages of each SLAM technique. It found that the filtering strategy, exemplified by the Extended Kalman Filter (EKF) and Unscented Kalman Filter (UKF), provides real-time estimating capabilities and computational efficiency. In nonlinear circumstances, it is vulnerable to linearization mistakes. The Rauch-Tung-Striebel (RTS) smoother and GraphSLAM, on the other hand, illustrate the smoothing technique, which improves accuracy by combining all available sensor readings and performing backward passes for refining. However, it requires more computer power and is better suited for offline processing.

The precise needs of the SLAM application determine whether to use a filtering or smoothing method. While smoothing methods excel in situations where greater accuracy is crucial and offline processing is practical, filtering approaches are helpful for real-time applications with constrained computational resources. Future research should concentrate on creating hybrid techniques that combine the advantages of

smoothing and filtering to increase accuracy and computing efficiency in a variety of SLAM applications.

References

1. Kolar, P., Benavidez, P., Jamshidi, M.: Survey of datafusion techniques for laser and vision based sensor integration for autonomous navigation. Sensors **20**(8), 2180 (2020)
2. Amjad, B., Ahmed, Q.Z., Lazaridis, P.I., Hafeez, M., Khan, F.A., Zaharis, Z.D.: Radio SLAM: A review on radio-based simultaneous localization and mapping. IEEE Access (2023)
3. Munguia, R., Trujillo, J.C., Obregón-Pulido, G., Aldana, C.I.: Monocular-based SLAM for mobile robots: filtering-optimization hybrid approach. J. Intell. Rob. Syst. **109**(3), 53 (2023)
4. Pu, H., Luo, J., Wang, G., Huang, T., Liu, H.: Visual SLAM integration with semantic segmentation and deep learning: a review. IEEE Sensors J (2023)
5. Jagadeeswari, M., Manikandababu, C.S., Aiswarya, M.: Integral images: efficient algorithms for their computation systems of speeded-up robust features (Surf). In: Pervasive Computing and Social Networking: Proceedings of ICPCSN 2021, pp. 663–672. Springer, Singapore (2022)
6. Li, Z., Zhao, N., Xiong, X., Yang, W., Wang, Z., Bie, X., Zou, X.: A graph optimization approach to range-based relative location. In: Journal of Physics: Conference Series, vol. 2591, no. 1, p. 012018. IOP Publishing (2023)
7. Pan, Y., Kompis, Y., Bartolomei, L., Mascaro, R., Stachniss, C., Chli, M.: Voxfield: Non-projective signed distance fields for online planning and 3d reconstruction. In: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5331–5338. IEEE (2022)
8. Lai, T.: A review on visual-SLAM: advancements from geometric modelling to learning-based semantic scene understanding using multi-modal sensor fusion. Sensors **22**(19), 7265 (2022)
9. Zhang, H., Ding, Z., Zhou, L., Wang, D.: Particle filtering SLAM algorithm for urban pipe leakage detection and localization. Wirel. Netw., 1–12 (2023)
10. Han, X., Yang, L.: SQ-SLAM: Monocular semantic slam based on superquadric object representation. J. Intell. Rob. Syst. **109**(2), 1–14 (2023)
11. Haddeler, G., Aybakan, A., Akay, M.C., Temeltas, H.: Evaluation of 3D LiDAR sensor setup for heterogeneous robot team. J. Intell. Rob. Syst. **100**, 689–709 (2020)
12. Zhang, S., Zhao, S., An, D., Liu, J., Wang, H., Feng, Y., Li, D., Zhao, R.: Visual SLAM for underwater vehicles: a survey. Comput. Sci. Rev. **46**, 100510 (2022)
13. Liu, J., Gao, Y., Jiang, X., Fang, Z.: Online object-level SLAM with dual bundle adjustment. Appl. Intell. **53**(21), 25092–25105 (2023)
14. Helgesen, H.H., Bryne, T.H., Wilthil, E.F., Johansen, T.A.: Camera-based tracking of floating objects using fixed-wing UAVs. J. Intell. Rob. Syst. **102**(4), 80 (2021)
15. Dhruv, P., Naskar, S.: Image classification using convolutional neural network (CNN) and recurrent neural network (RNN): a review. In: Machine Learning and Information Processing: Proceedings of ICMLIP 2019, pp. 367–381 (2020)
16. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B.: FastSLAM: A factored solution to the simultaneous localization and mapping problem. Aaai/iaai, 593598 (2002)
17. Martin, F., Dalphond, J., Tuck, N.: Teaching localization in probabilistic robotics. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 26, no. 3, pp. 2373–2374 (2012)
18. Klein, G., Murray, D.: Parallel tracking and mapping for small AR workspaces. In: 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, pp. 225–234. IEEE (2007)
19. Davison, A.J., Reid, I.D., Molton, N.D., Stasse, O.: MonoSLAM: Real-time single camera SLAM. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 1052–1067 (2007)

20. Mur-Artal, R., Montiel, J.M.M., Tardos, J.D.: ORB-SLAM: A versatile and accurate monocular SLAM system. IEEE Trans. Rob. **31**(5), 1147–1163 (2015)
21. Engel, J., Schöps, T., Cremers, D.: LSD-SLAM: Large-scale direct monocular SLAM. In: European Conference on Computer Vision, pp. 834–849. Cham: Springer International Publishing (2014)
22. Montemerlo, M., Thrun, S., Koller, D., Wegbreit, B.: A factored solution to the simultaneous localization and mapping problem. In: Proceedings of the National Conference on Artificial Intelligence, pp. 593–598 (2002)
23. Dellaert, F., Kaess, M.: Factor graphs for robot perception. Found. Trends Robot. **6**(1–2), 1–139 (2017)
24. Kaess, M., Johannsson, H., Roberts, R., Ila, V., Leonard, J.J., Dellaert, F.: ISAM2: Incremental smoothing and mapping using the Bayes Tree. Int J Robot. Res. **31**(2), 216–235 (2012)
25. Wen, S., Zhao, Y., Zhang, H., Lam, H.K., Manfredi, L.: Joint optimization based on direct sparse stereo visual-inertial odometry. Auton. Robot. **44**, 791–809 (2020)
26. Bergou, E.H., Diouane, Y., Kungurtsev, V.: Convergence and complexity analysis of a Levenberg–Marquardt algorithm for inverse problems. J. Optim. Theory Appl. **185**, 927–944 (2020)
27. Macario Barros, A., Michel, M., Moline, Y., Corre, G., Carrel, F.: A comprehensive survey of visual slam algorithms. Robotics **11**(1), 24 (2022)
28. Forster, C., Pizzoli, M., Scaramuzza, D.: SVO: Fast semi-direct monocular visual odometry. In: IEEE International Conference on Robotics and Automation, pp. 15–22 (2014)
29. Civera, J., Davison, A.J., Montiel, J.M.: Inverse depth parametrization for monocular SLAM. IEEE Trans. Rob. **24**(5), 932–945 (2008)
30. Guclu, O., Can, A.B.: Fast and effective loop closure detection to improve SLAM performance. J. Intell. Rob. Syst. **93**, 495–517 (2019)
31. Hammia, S., Hatim, A., Bouaaddi, A., Haijoub, A.: Lightweight hardware architecture of EKF-SLAM and its FPGA implementation. In: International Conference on Digital Technologies and Applications, pp. 743–752. Cham: Springer International Publishing (2022)
32. Gerlein, E.A., Díaz-Guevara, G., Carrillo, H., Parra, C., Gonzalez, E.: Embbedded system-on-chip 3D localization and mapping—eSoC-SLAM. Electronics **10**(12), 1378 (2021)
33. Contreras, L., Cruz, S., Motta, J.M.S., Llanos, C.H.: FPGA implementation of the EKF algorithm for localization in mobile robotics using a unified hardware module approach. In: 2015 International Conference on ReConFigurable Computing and FPGAs (ReConFig), pp. 1–6. IEEE (2015)
34. Bouhoun, S., Sadoun, R., Adnane, M.: OpenCL implementation of a SLAM system on an SoC-FPGA. J. Syst. Architect. **111**, 101825 (2020)
35. Ma, Z., Zhang, X.: FPGA-based sensorless control for PMSM drives using the stator/rotor frame extended Kalman filter. In: 2018 Chinese Control And Decision Conference (CCDC), pp. 102–107. IEEE (2018)
36. Xu, Z., Yu, J., Yu, C., Shen, H., Wang, Y., Yang, H.: CNN-based feature-point extraction for real-time visual SLAM on embedded FPGA. In: 2020 IEEE 28th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM), pp. 33–37. IEEE (2020)

Machine Learning to Predict Railway Infrastructure Defects

Khawla Elansari, Abdellah Idrissi, and Hajar Tifernine

Abstract The rapid advancement of the digital revolution has propelled machine learning (ML) to the forefront of transformative technologies within the transportation sector. However, exploring its potential for predictive maintenance in railway infrastructure is still burgeoning. This research paper delves into this nascent field, probing the trajectory of ML applications in railway maintenance and the effectiveness of various ML models in asset and equipment condition prediction. Employing a Literature Review (LR) and Systematic Literature Review (SLR)—which incorporates both the Delphi method and expert inputs alongside PRISMA-guided, AI-enhanced screening—this study aims to distill the influence of data quality, diversity, and specificity on ML efficacy in this context. The findings indicate a positive trend towards integrating ML, particularly highlighting the increasing reliance on artificial neural networks (ANN) and LSTM models for asset condition forecasting. Models like Random Forest and Gradient Boosting remain prevalent, underscoring the necessity to tailor approaches to the unique requirements of each railway component. The study underscores that while tracks are the primary focus of current research, significant potential lies in extending these methodologies to other critical assets, including signaling systems and catenaries. Data quality emerges as a pivotal factor; subpar data significantly impairs predictive accuracies, while data tailored to specific networks poses challenges to model generalizability. Identified gaps span data variability, model validation on actual datasets, parameterization, fault categorization, integration into maintenance planning, operational safety, and cost analysis. These insights call for concerted efforts to bridge these gaps, paving the way for robust, context-adaptable ML solutions in railway maintenance.

K. Elansari · A. Idrissi (✉) · H. Tifernine
Artificial Intelligence and Data Science Group, IPSS Team, Computer Science Laboratory (LRI), Computer Science Department, Faculty of Science of Rabat, Mohammed V University in Rabat, Rabat, Morocco
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_31

1 Introduction

In the continuously evolving landscape of transportation infrastructure, the railway sector stands out for its immense reliance on the seamless performance of its multi-faceted systems. The reliability and safety of railway operations hinge on the proactive management of infrastructure, where maintenance strategies play a pivotal role. The burgeoning generation of substantial data volumes within the railway sector, as documented by Kalathas and Papoutsidakis (2021), has catalyzed a paradigm shift towards predictive maintenance (PDM) strategies. The pivotal role of PDM emerges as it harnesses data analytics to pre-empt equipment failures, thus mitigating operational incidents that could result in financial burdens and compromise passenger safety, resource conservation, and customer satisfaction.

Railway infrastructure demands an agile and sustainable maintenance regime, including intricate subsystems like tracks, switches, embankments, catenaries, and signaling and telecommunication systems [1]. Varied equipment characteristics within these subsystems necessitate bespoke, data-informed maintenance plans. The advent of Machine Learning (ML), defined by [2] as the process of programming computers to optimize a performance criterion based on examples and past experiences, has propelled PDM into a new era. ML's predictive prowess facilitates accurate failure predictions and remaining useful life (RUL) estimations for equipment, marking a significant leap in maintenance strategy efficiency (Bukhsh & Stipanovic, 2020, [1].

Despite technological advancements, a mere 11% of companies have incorporated ML into their PDM practices, as Haarman, et al. (2017) indicated, hinting at a vast potential for its broader adoption. This paper aims to dissect ML-driven PDM strategies within railway maintenance, exploring suitable ML algorithms, data collection methodologies, the nature of railway equipment amenable to ML-assisted PDM, and the prevailing challenges. Through a rigorous literature review (LR), this research seeks to contribute to the systematic evaluation of current knowledge and practice.

The paper endeavors to address the following critical research questions:

RQ1: What is the trend regarding using Machine Learning (ML) in the predictive maintenance of railway infrastructure: which assets and equipment are concerned, which models are employed, and what performance can be observed?

RQ2: How do the quality, diversity, and specificity of data influence the effectiveness of ML models in the predictive maintenance of railway infrastructures?

RQ3: What are the main current limitations and gaps in applying Machine Learning to predictive maintenance in railway infrastructure?

The primary goal of this paper is to delineate the ML algorithms applied to railway infrastructure PDM, highlighting their potential applications, requisite inspection methods, and tools. It also aspires to classify data sources, outline relevant datasets, and specify the railway equipment under surveillance. Despite ML's potential in PDM, companies need help to harness these technologies fully within their internal

operations, prompting this study to identify current challenges in ML application to PDM.

Bound by the scope of ML concepts in railway infrastructure PDM, this paper excludes applications to other railway system components, like rolling stock. It focuses exclusively on failure prediction rather than anomaly detection. While it reviews general ML algorithm functions, it refrains from delving into their programming and implementation specifics.

In light of previous research, this paper builds upon the work presented in [1-63] among others, to fill gaps in the literature, particularly in the oversight of essential infrastructure components like signaling and catenaries. It also addresses the often-neglected aspect of operational safety in ML applications within railway PDM. By critically assessing past studies, this paper sets the stage for a comprehensive examination of ML's potential and limitations in transforming railway infrastructure maintenance for future operational resilience and reliability.

2 Methodology

In this research, we developed a comprehensive methodology consisting of a literature review (LR), a systematic literature review (SLR), and a quantitative analysis to delve into the complexities and diversity inherent in railway infrastructure. The LR served as the foundation for establishing a solid theoretical base. At the same time, the SLR was the primary method used in conjunction with the LR for data collection to address research questions. The quantitative analysis provided insights into general trends and data distributions, essential for evaluating the impact of categorical factors and the strength of relationships between reported variables.

The LR was conducted online, using primary and secondary keywords to extract relevant literature, and then qualitatively reviewed. The gathered theories are presented in later sections of the research.

For the SLR, we adhered to a systematic approach recommended by [64], which involves formulating clear and quantifiable research questions, devising a search strategy, selecting studies based on inclusion and exclusion criteria, critically evaluating the chosen studies, and synthesizing their findings. A preliminary search yielded 250 matches, but upon realizing that "railway infrastructure" was too specific, we used "railway" and "machine learning" on Scopus for an exhaustive search (Table 1).

Article selection was a two-pronged process. Initially, titles were scanned for relevance, followed by abstracts, and finally, a full-text review to ensure inclusion or exclusion. In parallel, we extracted a database from Scopus and reviewed it with AS Review, an AI tool for research document review. To ensure quality and relevance, we involved railway experts in the screening process, utilizing the Delphi method to refine expert judgment and integrate industry-specific standards.

The included articles were cataloged in an Excel file, recording pertinent information such as publication date, authors, journal, data collection methods, results, study limitations, machine learning models used, performance indicators, and Safety and

Table 1 Inclusion and exclusion criteria

Inclusion Criteria	Exclusion Criteria	Justification
Articles related to predictive maintenance (PDM), machine learning (ML), and the use of ML in PDM for railway infrastructure equipment	Documents unrelated to PDM, ML, or the use of ML in PDM for railway infrastructure equipment	The selected article must be related to using ML in PDM for railway infrastructure equipment. This is required to answer the stated research questions (RQs)
Presentation of models/ methods/techniques, including tests and results	Documents that present ideas/ hypotheses without any experimentation	Criteria are set to ensure that the proposed methods have been experimentally tested and the results are determined. The presented method is only helpful for this research if tested
Peer-reviewed articles	Grey literature (unpublished work and not peer-reviewed)	Peer-reviewed articles have the advantage of being published in scientific journals and have been subjected to critical review. This will enhance the quality assessment and reduce biased articles [65]
Articles published after 2014	Articles published before 2014	Due to the rapid evolution of information technologies (AI, ML) related to PDM, it has been decided to include only articles published after 2014
Primary sources	Secondary sources (interprets and comments on primary sources)	This SLR targets articles that involve original research and new findings and is therefore limited to primary sources
Documents in English and French	Documents in languages other than French or English	The research was limited to the English and French languages. While including other languages in the study could enhance the SLR, translation is a restrictive framework in a purely technical field (railway, ML, PDM)

cost–benefit analyses. This enabled a structured overview of the included articles, supporting further analysis.

Quality assessment of the articles involved evaluating their research design and execution, focusing on methodological quality to ensure reliable and valid sources. A checklist was employed to judge the quality based on objectives, research methodology, data collection, and results (Fig. 1).

A risk of bias was acknowledged, with efforts to mitigate it through methodological tools like a checklist. Despite the exhaustive search method, the potential for

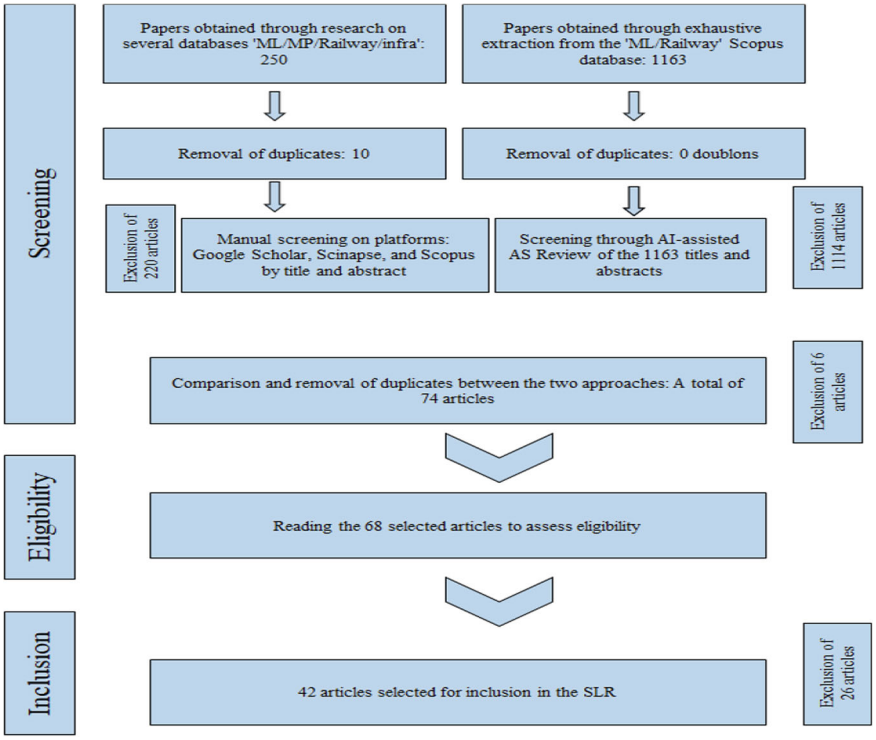


Fig. 1 Overview of the combined PRISMA methodology used

missing relevant articles remained, possibly affecting the SLR outcomes. However, the inclusion of only peer-reviewed articles and using two languages for research (English and French) aimed to minimize this bias.

The quantitative analysis utilized statistical methods to ensure the selected articles' relevance and analyze various parameters. Descriptive analysis provided an initial overview, while temporal analysis helped identify trends. ANOVA was used to compare group means, and correlation analysis determined the relationships between variables. These analytical methods allowed for exploring the diversity of data across the studies.

3 Results

Our systematic review meticulously evaluated a comprehensive dataset derived from a multi-stage screening process. Initially, 250 articles were identified through manual Google Scholar, Scinapse, and Scopus screening. We extracted an exhaustive database of 1163 articles related to ML applications in railways from

Scopus. Combining AI-assisted screening (ASreview Lab) and expert judgment, we employed a refined approach to manage the complexity and diversity of railway infrastructure components. This led to the exclusion of 220 articles based on titles and abstracts during manual screening and the elimination of 1114 articles through AI-assisted review. Subsequently, 68 articles were selected for thorough content review, culminating in 42 articles deemed suitable for inclusion in the systematic literature review (SLR), adhering to our research methodology's predefined inclusion and exclusion criteria.

3.1 Railway Assets

Railway infrastructure components such as track and geometry, signaling, and the catenary system are pivotal for train operations. Studies by Qing He, et al. (2022), Lei Han, et al. (2023), and Shuai Ma, et al. (2019) have focused on the tracks, highlighting their degradation and the importance of predictive maintenance. García-Sánchez, et al. (2020) explored the structural health of railway bridges using machine learning (ML) for proactive maintenance. Research into signaling assets, though less prevalent, is significant for traffic regulation and collision prevention, as indicated by Junyan Dai and Xiang Liu (2022) and Nielson Soares, et al. (2021). The catenary system, essential for electric trains, has been studied by Li Liu, et al. (2023) and Qi Wang, et al. (2020), with an emphasis on ML applications for predicting failures due to environmental interactions.

3.2 Railway Equipment

The rail thread is frequently studied due to its critical role in safety and stability, as seen in works by Lei Han, et al. (2023) and Shuai Ma, et al. (2019). The switch, another critical component, has been the focus of Jessada Sresakoolchai, et al. (2022) and Zaharah Allah Bukhsha, et al. (2019) for its importance in train routing. Railway bridges and catenary components, particularly for high-speed lines, have also been examined for predictive maintenance using ML.

3.3 Machine Learning Models

A variety of ML models are applied in predictive maintenance. Neural networks and their variants, such as the BDL-MLP by Qing He, et al. (2022) and CNNs by Jessada Sresakoolchai, et al. (2022), are prevalent. Random Forest and ensemble techniques are employed for their interpretability and nonlinear data handling, as utilized by Maximillian Weil, Negin Sadeghi, et al. (2022). RNNs and SVMs are effective for

time series data, as seen in the works of Lei Han, et al. (2023). Unsupervised methods like k-means and dimensionality reduction techniques are also utilized.

3.4 *Data Collection Methods and Sources*

Historical data, sensor data, inspection vehicles, and simulations are diverse data sources used in predictive maintenance research. Qing He, et al. (2022) and others have utilized these varied data collection methods to train and validate their ML models for effective predictive maintenance strategies.

3.5 *Quantitative and Statistical Analysis of Machine Learning Applications in Predictive Maintenance of Railway Infrastructures*

Predictive maintenance (PDM) stands at the forefront of revolutionizing railway infrastructure management. Integrating Machine Learning (ML) in PDM systems offers a proactive approach, promising enhanced reliability and safety. This section delves into a descriptive quantitative analysis based on data from the Railway Literature Survey (RLS), examining prevalent trends in ML applications specific to railway infrastructure maintenance (Fig. 2).

The word cloud generated from study titles reveals a lexicon centered around terms such as “railway,” “prediction,” “machine learning,” “data-driven,” “maintenance,” “track,” “geometry,” and “deep learning.“ The recurrent appearance of

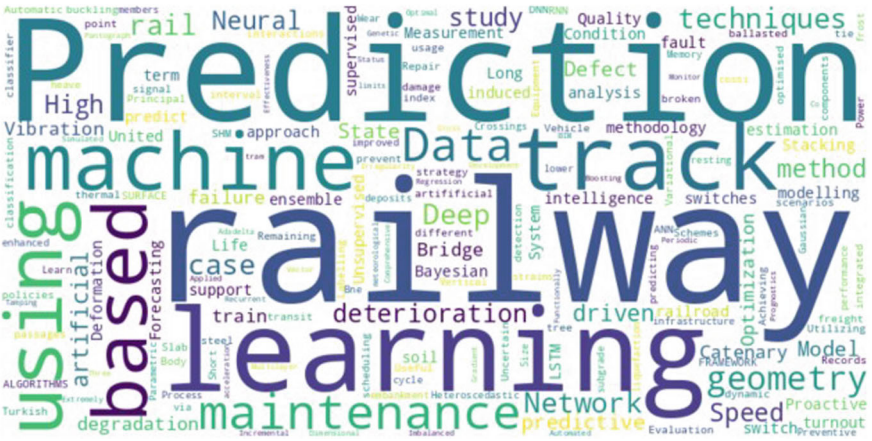


Fig. 2 Word Cloud of the 42 studies included

these terms underscores the research focus on employing ML techniques for predictive analysis and infrastructure upkeep, with a particular emphasis on the "track" asset. Further analysis indicates a growing propensity towards sophisticated neural networks, LSTM, and DNN models. The frequent references to "switches" and "turnouts" suggest a wide range of ML applications within the railway sector. Moreover, terms like "case study" imply a methodological approach grounded in empirical evidence, enhancing the practical relevance of the findings.

A deeper dive into the data reveals a predominant concentration of studies on tracks, marked by 34 occurrences, indicating a significant interest in applying PDM technologies to this specific infrastructure component. Catenaries, ranking second with four occurrences, and signaling, although less studied with two occurrences, reflect a diversity in the explored asset types. However, the disparity in focus suggests that essential aspects of railway infrastructure may be underrepresented in current research endeavors (Fig. 3).

ANNs emerge as the most frequently implemented ML model, with deep learning paradigms following suit. The occurrence rates of SVM, RF, and GB models are also noteworthy. This trend implies a research bias towards neural network-based models, likely due to their superior capacity for modeling complex data relationships. ANN, RF, and LSTM models are singled out for their superior performance in empirical studies (Fig. 4).

The "accuracy" metric dominates the performance evaluation landscape, with "precision" and "F-score" trailing behind. Despite this, the diversity of performance indicators suggests a multifaceted model assessment approach.

Sensors and inspection vehicles are the primary data collection tools, each with 11 occurrences, highlighting their importance in capturing direct and objective data. Historical data, with ten occurrences, underscores its significance in trend and pattern analysis. Conversely, simulations, database management systems, and inspection reports are less prevalent, indicating potential areas for methodological expansion (Fig. 5).

Using statistical methods to unpack variances in ML applications for predictive maintenance offers a lens through which to scrutinize the underlying data fabric of this research domain. The models displayed commendable precision levels, with accuracy averaging 0.8281 and a standard deviation of 0.1695. The concentration of

Fig. 3 Share of railway assets and equipment in publications

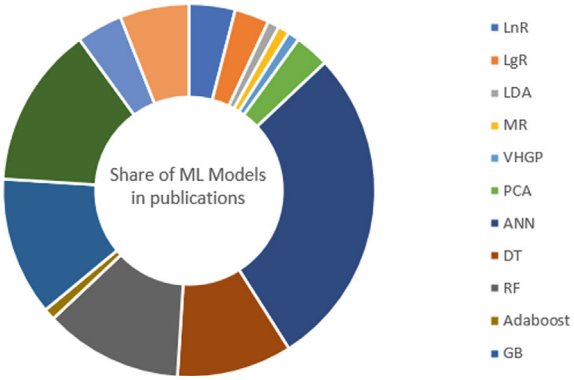


Fig. 4 Share of ML Models in publications

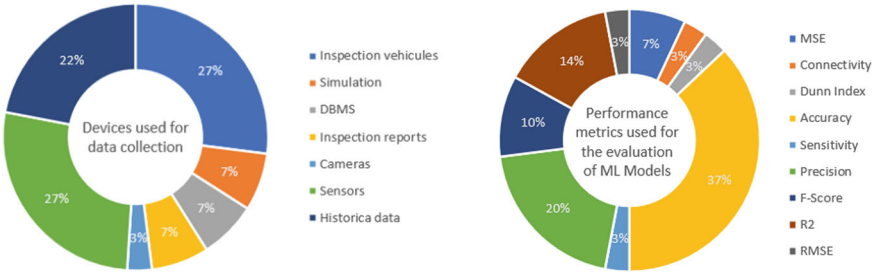


Fig. 5 Share of Data collection tools and metrics used in publications

data points around the 0.8 mark and a range stretching from 0.35 to 0.98 suggests a generally high efficacy of ML models in predictive maintenance applications—however, the variability indicated by the standard deviation points to an underlying heterogeneity in model performance (Fig. 6).

The ANOVA tests conducted reveal striking variances. For equipment, the F-value soars at 231,185.95 with a p-value at 0.00156, indicating significant variability in ANN model accuracies contingent on the equipment studied. This underscores the impact of equipment-specific characteristics on the predictability and, by extension, the precision of the models. On the data collection front, the F-value of 156.76 and a p-value of 0.05863, although not as dramatic as the equipment variance, still denote noteworthy differences in model accuracy across various data collection techniques. This variability can be attributed to the nature and quality of data these methods yield, impacting the model's ability.

Pearson's correlation analysis shows the relationship between model accuracies and the equipment or data collection techniques employed. A notable negative correlation with camera usage implies lower accuracy rates in ANN models, while positive

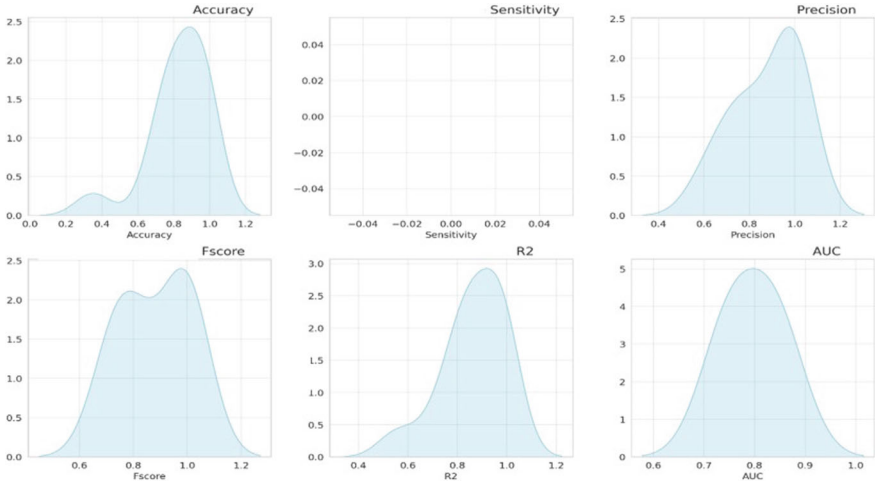


Fig. 6 Density curve of different metrics used in studies

correlations with sensors and simulation techniques suggest better predictive performance. These correlation coefficients, while insightful, are not definitive indicators of causality and warrant further investigation to establish robust conclusions.

4 Discussion

The analysis of publication trends and the ML models employed highlights an increasing adoption of advanced technologies for the predictive maintenance of critical railway infrastructure. This evolution reflects the continuous quest for operational efficiency and enhanced safety within the railway sector. However, certain assets, such as signaling systems, have received different attention than railway tracks or overhead lines, suggesting an unexploited potential for research and application.

Performance indicators such as Accuracy, Precision, and the F-score, used to evaluate ML models, have revealed that the rigorous evaluation of model performance is essential to ensure that ML models meet the necessary safety and reliability standards in the railway domain. The data collection process, comprising historical maintenance data, real-time sensor data, automated inspection vehicles, simulations, and inspection reports, forms a cornerstone for the research in predictive maintenance. However, the availability and quality of data remain a significant challenge that impacts the development and performance of predictive models.

The increasing utilization of Artificial Neural Networks (ANNs) and Long Short-Term Memory (LSTM) models is a notable trend, demonstrating a growing adoption of sophisticated techniques to predict the state of railway assets. Despite this, the variability in model accuracy across different equipment and data collection

methods emphasizes the need for tailored approaches to optimize outcomes for specific railway components. This variability also underscores the importance of considering environmental parameters, such as weather conditions, which have yet to be consistently integrated into predictive models.

The synthesis of the studies' strengths reveals several categories: diversity of models used, various equipment studied, data quality and diversity, and model performances. The diversity of ML models from classical to advanced deep learning techniques indicates a rich methodological landscape. The broad range of railway equipment studied, from railway tracks to overhead lines and signaling systems, illustrates the expansiveness of ML applications in predictive maintenance. High model performance in studies demonstrates the effectiveness of ML in this context. However, these positive outcomes must be balanced against the potential for overfitting and the challenge of generalizing models across different settings.

The limitations in applying ML to predictive maintenance are multifaceted. Data-related challenges include the quantity and quality of available data, which can limit model training and generalizability. Methodological limitations highlight issues with the generalizability and applicability of models to different regions and types of railway infrastructure. Practical limitations point to difficulties in distinguishing between different types of defects and integrating predictive maintenance findings into short-term planning and operations. Significantly, the operational safety of ML models in the railway context has yet to be exhaustively analyzed, presenting a crucial gap that future research must address.

The implications of the research, based on the 42 studies reviewed, are multifaceted and offer insightful perspectives for the future of railway infrastructure maintenance. These contributions include enhanced predictability and prevention of infrastructure failures, methodological diversification, optimization of resource allocation, safety improvement, the evolution of professional competencies, and the potential for future development and innovation in the field. The studies underscore the transformative potential of ML in the railway sector, particularly in predictive maintenance and infrastructure management, while also highlighting the challenges and opportunities associated with the adoption of these technologies.

5 Conclusion and Prospects

Our research delves into the transformative impact of Machine Learning (ML) on predictive maintenance within the railway sector, addressing the following research questions:

RQ1: ML Trends in Railway Infrastructure Maintenance.

We observed a notable trend toward adopting ML in predictive maintenance, with an uptick in deploying advanced models such as Artificial Neural Networks (ANNs) and Long Short-Term Memory (LSTM) networks. While these represent the cutting edge, models like Random Forest and Gradient Boosting maintain their relevance

due to their proven reliability. Railway tracks have emerged as the primary focus in existing literature, suggesting a vast untapped potential for other critical assets like signaling and overhead lines. Performance metrics vary considerably, with XGBoost models often delivering exceptional results. However, the efficacy of these models is tightly linked to the quality and specificity of the underlying data.

RQ2: Data Quality's Role in ML Effectiveness.

The nature of the data directly influences the success of ML models in predictive maintenance. High-quality, diverse, and asset-specific data enhances model accuracy, while poor-quality data can lead to unreliable predictions. The specificity of data to particular networks can limit model generalizability. We have found that integrating data from varied sources, including historical maintenance records, real-time sensors, automated inspection vehicles, and inspection reports, can bolster model robustness.

RQ3: Limitations and Gaps in ML Application.

Our analysis has surfaced several limitations in current ML applications. These include data quantity and quality issues, a lack of model generalizability across different assets, challenges in validating models on real-world data sets, and difficulties in fine-tuning model parameters. Additionally, the industry needs help integrating predictive maintenance insights into operational planning, ensuring system safety, and performing comprehensive cost–benefit analyses.

Moving forward, the studies pinpoint significant advancements and untapped potential for ML in railway predictive maintenance. To overcome current limitations, future research should explore:

- In-Depth Safety Analysis: Further research is needed to assess the operational safety of ML models, focusing on the implications of predictive errors on overall system and passenger safety.
- Cost–Benefit Evaluation: In-depth studies should be conducted to compare the financial implications of ML-based predictive maintenance with traditional maintenance approaches, considering implementation, operation, and maintenance costs against potential savings.
- Development of Global Models: Future studies might consider creating universal models capable of spanning various railway assets, promoting an integrated predictive maintenance approach.
- Establishment of a Global Database: A worldwide database, possibly spearheaded by international railway organizations, would be a significant step toward standardizing and harmonizing data for ML model training.
- Optimization and Validation of Models: Further exploration and validation of ML models and algorithms are essential to enhance performance and ensure reliability and applicability in real-world operational environments.
- Multi-Source Data Integration: Integrating and analyzing data from diverse sources will improve prediction accuracy.
- Data Standardization and Harmonization: Additional efforts for standardizing data across different sources and systems will facilitate the creation of common standards for railway data.

Addressing these areas will fill current gaps and contribute to the successful evolution and adoption of ML technologies in the railway sector, optimizing safety, reliability, and efficiency worldwide [63, 66–79].

References

1. Matic, A.: Mise en oeuvre de la maintenance prédictive dans le secteur ferroviaire Maintenance des infrastructures Trondheim: NTNU (2021)
2. Alpaydin, E.: Machine learning: the new AI. The MIT Press, Cambridge, Massachusetts (2016)
3. Xie, J., Huang, J., Zeng, C., Jiang, S.-H., Podlich, N.: Systematic literature review on data-driven models for predictive maintenance of railway track: Implications in geotechnical engineering. Geosciences (Basel) **10**, 1–24 (2020)
4. Kasraei, A., et al.: Optimal track geometry maintenance limits using machine learning: A case study. In: Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit (2020)
5. Lasisi, A., et al.: Machine Learning Ensembles and Rail Defects Prediction: Multilayer Stacking Methodology. ASCE-ASME J. Risk Uncertain. Eng. Syst., Part A: Civ. Eng. (2019)
6. Lasisi, A., Attoh-Okine, N.: Principal components analysis and track quality index: A machine learning approach. Elsevier Transp. Res. (2018)
7. Alsharif, M. H., Kelechi, A. H., Yahya, K. Chaudhry, S. A.: Machine learning algorithms for intelligent data analysis in the Internet of things environment: Taxonomies and research trends. Symmetry (Basel), pp. 12, 88 (2020)
8. Alzubi, J., Nayyar, A. Kumar, A.: Machine Learning from Theory to Algorithms: An Overview. J. Phys.: Conf. Ser. (2018)
9. Dutta, A., Kamaljyoti Nath Learning via Long Short-Term Memory (LSTM) network for predicting strains in Railway Bridge members under train induced vibration Lecture Notes in Electrical Engineering book series (LNEE, vol. 783) (2021)
10. Falamarzi, A., et al.: Development of a tram track degradation prediction model based on the acceleration data. Struct. Infrastruct. Eng. Maint., Manag., Life-Cycle Des., Perform. **15**, 2019 (2019)
11. Tiryaki, A.: Prediction of railway switch point failures by artificial intelligence methods. Turk. J. Electr. Eng. Comput. Sci. (2020)
12. Baloglu, O., Latifi, S. Q., Nazha, A.: What is machine learning? Arch Dis Child Educ Pract Ed, edpract-2020–319415 (2021)
13. Benmansour, M.A., Laroche, E., Benhaddou, S.: Sûreté de fonctionnement des systèmes ferroviaires utilisant l'intelligence artificielle : état de l'art et perspectives. Revue internationale des transports **52**(3), 355–372 (2022)
14. Goodman, K.E., Kaminsky, J., Lessler, J.: What is Machine Learning? A Primer for the Epidemiologist. Am. J. Epidemiol. **188**, 2222–2239 (2019)
15. Burkov, A. The Hundred-Page Machine Learning Book, Kindle Direct 73 (2019)
16. Vale, C., et al.: Prediction of Railway Track Condition for Preventive Maintenance by Using a Data-Driven Approach MDPI Infrastructures (2022)
17. Wei Tan1, C., et al.: Tamping Effectiveness Prediction Using Supervised Machine Learning Techniques Proceedings First International Conference on Rail Transportation 2017 (2018)
18. Ngamkhanong, C., Kaewunruen2, S.: Prediction of thermal-induced buckling failures of ballasted railway tracks using Artificial 1 Neural Network (ANN). Int. J. Struct. Stab. Dyn. (2022)
19. Claessens, M., Larochelle, S., Van den Heuvel, W.J.: Safety of artificial intelligence in railway systems: A survey of methods and tools. IEEE Trans. Intell. Transp. Syst. **23**(1), 279–296 (2022)

20. Cohen, J., Cohen, P., West, S.G., Aiken, L.S.: Applied multiple regression/correlation analysis for the behavioral sciences. Lawrence Erlbaum Associates (2003)
21. Hovad, E., et al.: Deep Learning for Automatic Railway Maintenance. Springer Series in Reliability Engineering (2021)
22. García-Sánchez, et al.: Gradient-Boosting Applied for Proactive Maintenance System in a Railway Bridge. Eur. Work. Struct. Health Monit. (2020)
23. Vassos, G., et al.: Labelling the State of Railway Turnouts Based on Repair Records. Springer Series in Reliability Engineering (2021)
24. Guler, H.: Prediction of railway track geometry deterioration using artificial neural networks: a case study for Turkish state railways. Maint. Manag. Life-Cycle Des. Perform. **10**(2014) 2014
25. Khajehei, H., et al.: Prediction of track geometry degradation using artificial neural network: a case study. Int. J. Rail Transp. (2021)
26. Hastie, T., Tibshirani, R., Friedman, J.H.: The elements of statistical learning. Springer, New York (2009)
27. Higgins, J.P.T., Green, S. (Eds.). Cochrane handbook for systematic reviews of interventions (Version 5.1.0). The Cochrane Collaboration (2011)
28. Cárdenas-Galloa, I., et al.: An ensemble classifier to predict track geometry degradation. Elesivier Reliab. Eng. Syst. Saf. (2017)
29. James, G., Witten, D., Hastie, T., Tibshirani, R.: An introduction to statistical learning. Springer, New York (2013)
30. Sresakoolchai, J., et al.: Track Geometry Prediction Using Three-Dimensional Recurrent Neural Network-Based Models Cross-Functionally Co-Simulated with BIM. MDPI sensors (2022)
31. Sresakoolchai, J., et al.: Prediction of turnout support deterioration through dynamic train-track interactions integrated with artificial intelligence Inter noise University of Birmingham (2022)
32. Chen, J., et al.: A deep learning forecasting method for frost heave deformation of high-speed railway subgrade. Elsevier Cold Reg.Ns Sci. Technol (2021)
33. Sainz-Aja, J.A., et al.: Parametric analysis of railway infrastructure for improved performance and lower life-cycle costs using machine learning techniques. Elsevier Adv. Eng. Softw. (2023)
34. Lee, J.S., et al.: Prediction of track deterioration using maintenance data and machine learning schemes. J. Transp. Eng. (2018)
35. Lee, J.S., et al.: Deterioration prediction of track geometry using periodic measurement data and incremental support vector regression model. ASCE J. Transp. Eng. (2019)
36. Dai, J., Liu, X.: Machine learning based prediction of rail transit signal failure: A case study in the United States. Sage J.S (2022)
37. Grace Mercy, K., Sri. K., Rao, S.: A Framework for Rail Surface Defect Prediction using Machine Learning Algorithms. In: 2018 International Conference on Inventive Research in Computing Applications (ICIRCA) (2018)
38. Han, L., Liao, Y., Wang, H., Zhang, H.: Long-term prediction for railway track geometry based on an optimized DNN method. Elsevier (2023)
39. Liu, L., et al.: Remaining Useful Life Prediction for a Catenary, Utilizing Bayesian Optimization of Stacking. MDPI Electronics (2023)
40. Marhon, S.A., Cameron, C.J.F., Kremer, S.C., Bianchini, M., Maggini, M., Jain, L. C.: Recurrent Neural Networks, Berlin, Heidelberg (2013)
41. Weil, M., Sadeghi, N., et al.: Machine learning based predictive modeling of a steel railway bridge for damage modeling of train passages and different usage scenarios. In: European Workshop on Structural Health Monitoring Conference paper 2022
42. McKinsey AND Company. The promise of predictive maintenance in the rail industry (2022)
43. Meindl, B., Ayala, N. F., Mendonça, J., Frank, A.G.: The four smarts of Industry 4.0: Evolution of ten years of research and future perspectives. Technol. Forecast. Soc. Chang. 168, 120784 (2021)
44. Murphy, K.P.: Machine learning: A probabilistic perspective. MIT Press, Cambridge (2012)
45. Soares, N., et al.: Unsupervised machine learning techniques to prevent faults in railroad switch machines. Elsevier Int. J. Crit.Al Infrastruct. Prot. (2021)

46. Lopes Gerum, P. C., et al.: Data-driven predictive maintenance scheduling policies for railways. Elsevier Transp. Res. (2019)
47. Wang, Q., et al.: Achieving Predictive and Proactive Maintenance for High-Speed Railway Power Equipment with LSTM-RNN. IEEE Trans. Ind. Inform. (2020)
48. Wang, Q., et al.: Measurement and Forecasting of High-Speed Rail Track Slab Deformation under Uncertain SHM Data Using Variational Heteroscedastic Gaussian Process MDPI Sensors (2019)
49. He, Q., Sun, H., Dobhal, M., Li, C., Mohammadi, R.: Railway tie deterioration interval estimation with Bayesian deep learning and data-driven maintenance strategy. Elsevier (2022)
50. Lin, S., et al.: A fault prediction method for catenary of high-speed rails based on meteorological conditions. J. Mod. Transp. (2019)
51. Ma, S., et al.: Deep Learning for Track Quality Evaluation of High-Speed Railway Based on Vehicle-Body Vibration Prediction. IEEE (2019)
52. Shumway, R.H., Stoffer, D.S.: Time series analysis and its applications. Springer (2011)
53. Siddaway, A.P., Wood, A.M., Hedges, L.V.: How to do a systematic review: A best practice guide for conducting and reporting narrative reviews, meta-analyses, and meta-syntheses. Annu. Rev. Psychol. **70**, 747–770 (2019)
54. Sharma, S., et al.: Data-driven optimization of railway maintenance for track geometry. Elsevier (2018)
55. Kocbek, S. Gabrys, B.: Automated machine learning techniques in prognostics of railway track defects. In: IEEE 2019 International Conference on Data Mining Workshops (ICDMW) (2019)
56. Jessada, S., Sakdirat, K.: Railway defect detection based on track geometry using supervised and unsupervised machine learning. Struct. Health Monitoring—Sage J.S (2022)
57. Ghani, S., Kumari, S.: Prediction of soil liquefaction for railway embankment resting on Bne soil deposits using enhanced machine learning techniques. J. Earth Syst. Sci. (2023)
58. Najeh, T., et al.: Deep-Learning and Vibration-Based System for Wear Size Estimation of Railway Switches and Crossings. MDPI Sensors (2021)
59. Xiao, C., Sun, J.: Convolutional Neural Networks (CNN). Introduction to Deep Learning for Healthcare. Cham: Springer International (2021a)
60. Wang, X., et al.: A machine learning based methodology for broken rail prediction on freight railroads: A case study in the United States. Elsevier Constr. Build. Mater. (2022)
61. Chen, Y., et al.: Learn to predict vertical track irregularity with extremely imbalanced data. Proc. Mach. Learn. Res. (2021)
62. Allah Bukhsha, Z., et al.: Predictive maintenance using tree-based classification techniques: A case of railway switches. Elsevier Transp. Res. (2019)
63. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
64. Moher, D., Liberati, A., Tetzlaff, J., Altman, D. G., and the PRISMA Group: Preferred reporting items for systematic reviews and meta-analyses: The PRISMA statement. Ann. Intern. Med. **151**(4), 264–269 (2009)
65. Purssell, E., Mccrae, N.: How to Perform a Systematic Literature Review: A Guide for Healthcare Researchers. Springer International Publishing, Practitioners and Students, Cham, Cham (2020)
66. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Min. (2017)
67. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks. arXiv preprint arXiv:1307.5910 (2012)
68. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
69. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
70. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and Skyline for Cloud Services Research and Selection System. Int. Conf. Big Data Adv. Wirel. Technol. (2016)

71. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF, 107–116 (2006)
72. Abourezq, Idrissi A.: A Cloud Services Research and Selection System. IEEE ICMCS (2014)
73. Abourezq, M, Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
74. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
75. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2021) 2020
76. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)
77. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
78. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless technologies (2016)
79. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
80. Zhijian Q.U., et al.: Genetic Optimization Method of Pantograph and Catenary Comprehensive Monitor Status Prediction Model Based on Adadelta Deep Neural Network. IEEE (2019)
81. Elhandri, K., Idrissi, A.: Comparative study of Top_k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secur. Trans. **10** (2020)

Discovered Process-Aware IoT Models Through Semantic Enrichment

El Kodssi Iman and Sbai Hanae

Abstract Business process management (BPM) is a popular approach that has attracted the attention of many researchers all through the years. BPM has evolved into Business Process Intelligence to provide more intelligence to the management of business processes. Process mining is one of the newest subfields of business process intelligence, which looks for, automatically, and validates and improves business processes. Given the importance of the Internet of Things, we presented contributions aimed at improving business process management in an IoT environment. In this context, the absence of modeling notions that define Internet of Things elements as parts of a business process model is clearly an important obstacle to autonomous business process detection. In this study, we provided an expanded BPMN ontology model for IoT that served as the foundation for a semantic framework.

Keywords Internet of Things (IoT) · Business process management notation ontology (BPMNO) · Process mining (PM)

1 Introduction

All businesses have recently relied on process logic to accomplish their business goals. Process logic is the interaction of hardware, software, people, and environment to achieve a predefined business goal. This process logic was generated by a process engine because of the interest in analyzing data collected by Internet of Things (IoT) devices to investigate human behavior. The Internet of Things (IoT), which is a network of intelligent equipment and sensors, may be used as the foundation for 2 this process management. In this instance, the IoT and the process engine collaborate to manage processes in a smart environment. (1) Although process mining (PM) and

E. K. Iman (✉) · S. Hanae
Mathematics, Computer Science and Application Laboratory FST Mohammedia, Hassan II University of Casablanca, Casablanca, Morocco
e-mail: elkodssi.iman@gmail.com

S. Hanae
e-mail: hanae.sbai@univh2c.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_32

smart spaces have both seen an accelerated phase of development as distinct fields of study in recent years, academics have lately begun investigating the pairing of the two, with some interesting outcomes. Human behaviors may be represented and seen as processes thanks to the use of PM methods on data from smart spaces [1]. However, even though process models may be recovered from smart space data, the processes developed using process mining approaches are not IoT, according to a lot of research published in the first articles [2–4]. In this regard, there are several methods for transforming sensor data into an event record. In our first research, we suggested a model-driven architecture to convert IoT data into event log format. and the purpose of this study is to semantically annotate this event log by using an enrichment strategy that incorporates both the expertise we've amassed via the addition of ideas from the business domain and the intelligent environment to the BPMNO, which specifies the structure of event logs. The main contribution of our work is to first provide an overview and comparison of semantic PM techniques applied in different domains and to smart spaces in order to analyze how these techniques currently approach the concept of semantic PM and subsequently propose a framework that allows for automatic generation of an IoT-aware process.

The remainder of this article is structured as follows: The next section introduces some basic concepts and terminology commonly used in the fields of smart spaces and process mining. Section 3 describes related work in semantic process mining. The results are reported in Sect. 4. Section 5 concludes the article with an overview of the main findings.

2 Basic Concepts

In this section, we define two key terms from our research:semantic in event logs and process-aware Internet of Things discovery.

2.1 Process-Aware Internet of Things Discovery

The Internet of Things (IoT) is a network of connected devices that exchange and gather data over the Internet. With contrasting and different requirements, the IoT encompasses a wide range of application areas [5, 6]. In the context of business process management in an intelligent environment, according to [7, 8], process-aware IoT is a connection between BPs and internet of things gadgets that intends to watch BPs and take suitable action about them. Providing all necessary IoT data for the process is the aim of an IoT-enabled business process model. Therefore, 3 collecting and evaluating sensor data is necessary in order to carry out business processes. To discover a model of this type, automatically discover IoT-aware business process models. Process mining techniques are used. The term "process mining" refers to a developing area of study that works with a number of distinct tasks that are all

connected by the common objective of learning from the available log files. IoT-aware process models are automatically found utilizing process mining techniques based on event logs.

2.2 Semantics in the Event Log

Event logs include a summary of details regarding how a process was carried out, including the time of the event, the case study, the activity, and the resources used [9]. As it facilitates the discovery, verification, and improvement of business process models, this real-world data is essential to the field of process mining. MXML (Mining eXtensible Markup Language) and XES (eXtensible Event Stream) are two frequently used formats for presenting and saving these logs [10] . To support semantic annotations related to ontological concepts, XES uses the "Semantic" extension.2

3 Related Work

In this section, we review the most recent and important studies in the literature on semantic process mining approaches. We compare these works according to their contributions, in terms of Process Mining Area, Input log, language, Annotated element, Type of Ontologies and Domain of application. KINGSLEY OKOYE made one of the first surveys of our study in [12]. This study illustrates how the semantic concepts can be arranged on top of the generated models in order to provide a more contextual review of the models created using the conceptualization approach. In order to enable a more abstract analysis of the extracted logs or model, the method formally entails semantic annotation of the process elements with concepts that they represent in real-time settings and linking them to an ontology. As a result, the resulting models are more informative. In [14] the association of semantic annotations to educational event logs has been shown to enable the extraction of more precise and condensed educational processes, according to Auteur. Additionally, it provides a semantic matching method that (semi-)automatically connects educational labels to the pertinent concepts of an educational ontology. The approach [15] was proposed by the authors as a means of semantically improving a shared knowledge base of process models used by "Business Process as a Service" (BPaaS) providers. Their goal is to use this knowledge base to find suitable process fragments for particular positions in a business process that can be suggested to make the modeling of process 4 variants easier. In [16] authors present a general strategy for enhancing process mining by utilizing event logs connected to ontology structures. This method's ability to evaluate and process models at various levels of abstraction is what makes it interesting. More recently, an approach that addresses semantic process mining in the IoT context has been proposed [13, 11]. The authors of this study [13] have

developed an extension to XES that makes it easier to combine process event logs with IoT data related to the environment. This extension makes it possible to use process mining techniques as well as visualization tools that have been specially designed for IoT event logs. From a different angle, the author proposal for [11] entails using domain-specific ontologies to achieve device and service abstraction, enabling process mining and service orchestration analysis within heterogeneous IoT environments. A prototype IoT environment and mining methods created within the Process Mining Platform (ProM) were used to evaluate this strategy. To recapitulate, Table 1 presents a comparison of the related works presented in this section based on the following criteria:

- Process Mining Area: It indicates the category of process mining techniques (discovery, conformance, or enhancement) studied in the work.
- Input log: this is data from various sources in an intelligent environment or another environment that is populated to enable the application of process mining techniques.
- The modeling languages used are: Most of the approaches concern the ontology domain and format of data input; on the one hand, Event Log Language indicates the language used to represent event logs.
- On the other hand, ontology language designates the language used to represent ontologies.
- Annotated element: refers to the element that has been annotated using the ontology (task or resource).
- Type of Ontologies: determines the type of ontology used.
- Domain ontology application: In order to represent domain constraints in business process modeling, different approaches introduce a domain ontology that captures real-world concepts.

A comparison of these works shows us that there are few that focus on semantic log enrichment for the automatic management of an IoT-aware process model. Furthermore, the work of DataStream XES Extension: Embedding IoT Sensor Data into Extensible Event Stream Logs and Semantics Based Service Orchestration in IoT is limited to enriching via the semantic annotation of TASK ELEMENT [11, 15, 16] to extract an IoT-enriched event log. This, existing approaches are not sufficient to prepare an IoT-enriched event log for the purpose of automatically generating an IoT-aware process model. It is therefore necessary to integrate all perspectives when semantically enriching event logs. In addition, existing approaches [12, 14] use domain ontologies, which presents a gap for the discovery of IoT-aware process models given that this type of model must handle IoT elements. Existing work uses OWL (Web Ontology Language) and WSML (Web Service Modeling Language) to represent ontologies and uses event logs expressed in XES (eXtensible Event Stream) or MXML (XML-based markup language for the user interface).

When analyzing existing approaches, we note that most of them are limited to the semantic annotation of activity elements using a domain ontology. These approaches appear to be unsuitable for the preparation of IoT-enriched event logs in the domain of

Table 1 Comparative study

Approche	Process Mining Area	Support Embedded IoT	Input	Langage		Element Annotated	Basic ontologies	Domain of ontologies application
				Input data	Ontologies			
[11]	Discovery	Yes	Collected from sensors Commands through various user	***	FaCT + + SSN OSGi	Task	Semantics in IoT Domain ontology	Service Orchestration
[12]	Enhancement	NO	Events Log Process Models	***	OWL	Task	Domain ontology	Learning Domain
[13]	Discovery	Yes	Data IoT Process event	XES	***	Process Task	Extension data stream XES Domain ontology	Transport logistics and manufacturing
[14]	Discovery	NO	Event log Process Models	MXML	WSML	Task	Domain ontology	Education context
[15]	Enhanced	NO	Events log Process Models	XES	OWL BPMO NCFO	Task	Process model ontology	The concept of variability
[16]	Enhanced	NO	Event log Process Models	***	Process model ontology Domain ontology	Task Ressource	Process model ontology Domain ontology	***

process mining models, as we need to enrich the event log with domain concepts and IoT concepts. For these reasons, we propose an approach to semantically annotating an event log with an IoT domain association ontology and a domain ontology. Then, process mining techniques will be applied to IoT-enriched event logs to discover semantically annotated IoT-aware process models.

4 Approach Overview

In a previous publication, we presented a framework for generating model processes in an intelligent environment. Our A previous publication [17] provides a simplified illustration of the proposed framework (Fig. 1). In this publication, we focus more on the semantic process mining and IoT context parts, i.e., how to semantically annotate an event log via a proposed ontology. Figure 1 represents a simplified illustration of the proposed framework.

For more details:

- Step 1*:* Semantic annotation of the process model by the ontology "extended BPMNO for IoT": this component takes an event log as input, and we obtain an event enriched by the IoT element. We have provided an overview of the original project [18].
- Step 2*:* Process model discovery: this component takes as input the event log enriched by the IoT element and applies process mining techniques to discover the process model-aware IoT, which is semantically enriched with the same ontologies cited above as well as the rules we'll be developing in future publications.

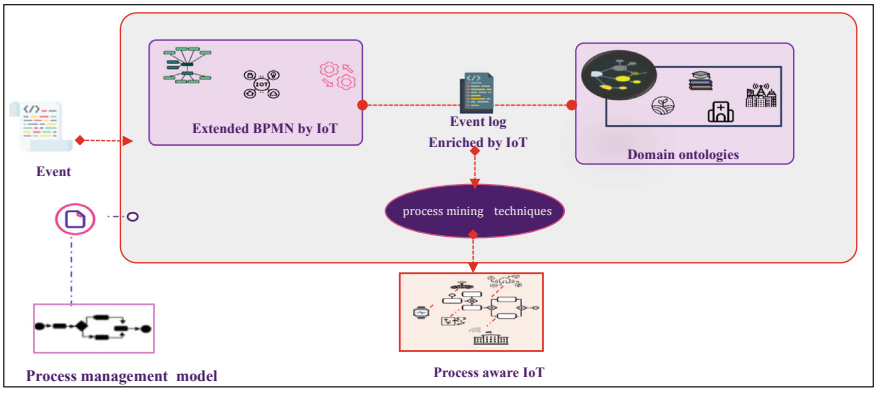


Fig. 1 Framework for automatic discovery of semantically enriched process aware IoT model

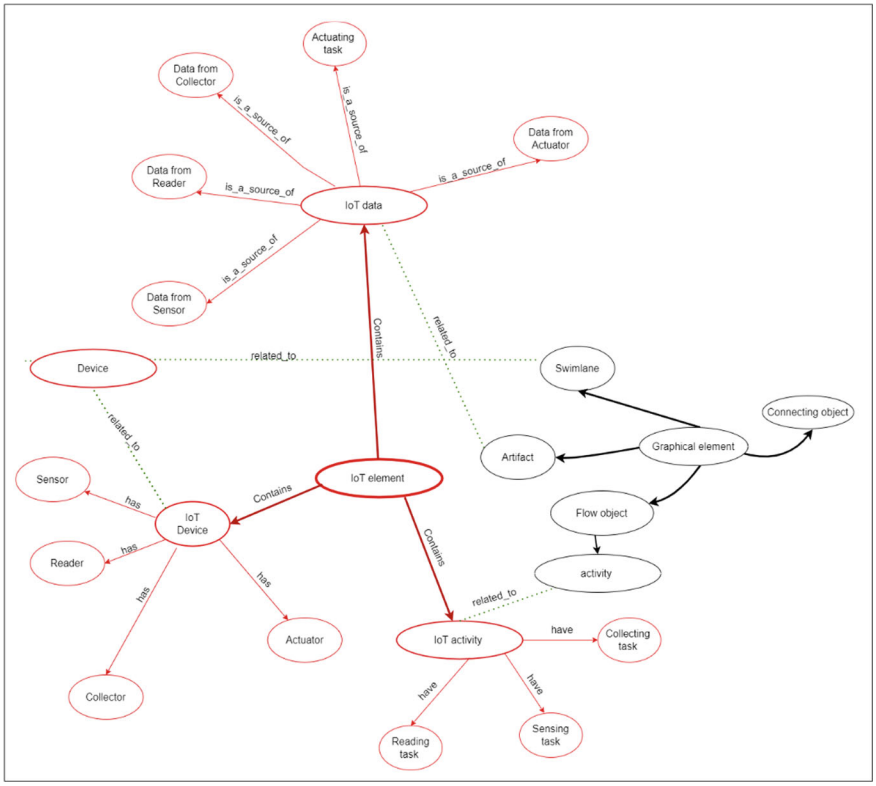


Fig. 2 Extending Business Process Modeling Ontology by IoT element

5 Conclusion

A wide variety of linked gadgets make up intelligent surroundings. It opens up new possibilities but also increases the complexity of business process management. More studies have concentrated on discovering business process models, using PM, and converting IoT data into event logs. But a comparison analysis revealed that this approach lacks components related to the Internet of Things. In this study, we have suggested an extended PBMN-based semantic framework for the IoT model. We may semantically improve and enrich the business processes that PM identifies by linking IoT ideas to business process concepts. As part of our continuous work, we will develop the Extended BPMN for IoT ontology and implement the framework.

References

1. Cook, D.J., Krishnan, N.C.: Activity Learning: Discovering, Recog-nizing, and Predicting Human Behavior from Sensor Data. John Wiley & Sons (2015)
2. Banziger, R.B., Basukoski, A., Chaussalet, T.: Discovering Business Processes in CRM Systems by Leveraging Unstructured Text Data, In: 2018 IEEE 20th International Conference on High Performance Computing and Communications; IEEE 16th International Conference on Smart City; IEEE 4th International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 1571–1577 (2018). https://doi.org/10.1109/HPCC/SmartCity/DSS.2018.00257
3. Saouli, R.A. Benhassine, N.: La ville intelligente, une stratégie pour un développement urbain durable. JAEC **6**(2), 64–76 (2021)
4. Burhanuddin, M.A., Mohammed, A.A.-J., Ismail, R., Basiron, H.: In-ternet of Things Architecture: Current Challenges and Future Direction of Re-search, **12**(21) (2017)
5. Finkenzeller, K.: RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication. John Wiley & Sons (2010)
6. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): A vision, architectural elements, and future directions. Futur. Gener. Comput. Syst. **29**(7), 1645–1660 (2013). https://doi.org/10.1016/j.future.2013.01.010
7. Rathee, G., Khelifi, A., Iqbal, R.: Intelligent Data Management Techniques in Multi-Homing Big Data Networks (2021). https://doi.org/10.1155/2021/5754322
8. An IoT and business processes based approach for the monitoring and control of high value-added manufacturing processes|Proceedings of the International Conference on Future Networks and Distributed Systems (2023). https://doi.org/10.1145/3102304.3102341
9. Allani, O., Ghannouchi, S.A.: Verification of BPMN 2.0 process models: An event log-based approach. Procedia Comput. Sci. **100**, 1064–1070 (2016). https://doi.org/10.1016/j.procs.2016.09.282
10. Verbeek, H.M.W., Buijs, J.C.A.M., van Dongen, B.F., van der Aalst, W.M.P.: XES, XESame, and ProM 6, In: Information Systems Evolution, P. Soffer et E. Proper, (eds.), In: Lecture Notes in Business Information Processing. Berlin, Heidelberg: Springer, pp. 60–75 (2011). https://doi.org/10.1007/978-3-642-17722-4_5
11. Okoye, K., Islam, S., Naeem, U., Sharif, S.: Semantic-Based Process Mining Technique for Annotation and Modelling of Domain Processes
12. Awatef, H., et al.: Using Semantic Lifting for Improving Educational Process Models Discovery and Analysis, vol. 1293 (2014)
13. Yongsiriwit, K., Sellami, M., Gaaloul, W.: Semantic process fragments matching to assist the development of process variants: In: 2015 IEEE International Conference on Services Computing, pp. 712–719 (2015). https://doi.org/10.1109/SCC.2015.101
14. Nykänen, O., Rivero-Rodriguez, A., Pileggi, P., Ranta, P.A., Kailanto, M., Koro, J.,: Associating event logs with ontologies for semantic process mining and analysis. In: Proceedings of the 19th International Academic Mindtrek Conference, Tampere Finland: ACM, pp. 138–143 (2015). https://doi.org/10.1145/2818187.2818273
15. Future Internet|Free Full-Text|DataStream XES Extension: Em-bedding IoT Sensor Data into Extensible Event Stream Logs (2023). https://www.mdpi.com/1999-5903/15/3/109
16. Chindenga, E., Scott, M.S., Gurajena, C.: Semantics based service orchestration in IoT. In: Proceedings of the South African Institute of Computer Scientists and Information Technologists, in SAICSIT '17. New York, NY, USA: Association for Computing Machinery, pp. 1–7 (2017). https://doi.org/10.1145/3129416.3129438
17. Iman, E., Laanaoui, M.D., Sbai, H.: Applying process mining to sensor data in smart environment: A comparative study. In: Innovations in Smart Cities Applications Volume 6, M. Ben Ahmed, A. A. Boudhir, D. Santos, R. Dionisio, et N. Benaya, (eds.), In: Lecture Notes in Networks and Systems. Cham: Springer International Publishing, pp. 511–522 (2023). https://doi.org/10.1007/978-3-031-26852-6_47

18. Elkodssi, I, Sbai, H.: Toward a new semantic framework for internet of things-aware business process discovery. ITM Web Conf., vol. 52, p. 02001 (2023). https://doi.org/10.1051/itmconf/20235202001

Predicting Credit Risk of SMEs in Malaysia: Machine Learning vs Deep Learning

Syahida Abdullah and Roshayu Mohamad

Abstract This study helps in predicting the credit risk of small and medium-sized (SMEs) by developing models using artificial intelligence algorithms: Machine Learning (ML) and Deep Learning (DL). This is demonstrated using four different prediction algorithms—Random Forest (RF), Neural Network (NN), Support Vector Machine (SVM) and Decision Tree (DT). The results show that DL model produced better prediction result: 86.5% classification accuracy, 86.1% of F1 score for Random Forest Model; 85.3% of classification accuracy, and 84.3% of F1 score for Neural Network Model. The result confirms that DL algorithm provides more accurate prediction leading to higher accuracy.

1 Introduction

1.1 Scope and Purpose

This study helps in predicting the credit risk of small and medium-sized (SMEs) by developing models using artificial intelligence algorithms: Machine Learning (ML) and Deep Learning (DL). This is demonstrated using four different prediction algorithms—Random Forest (RF), Neural Network (NN), Support Vector Machine (SVM) and Decision Tree (DT). The results show that DL model produced better prediction result: 86.5% classification accuracy, 86.1% of F1 score for Random Forest Model; 85.3% of classification accuracy, and 84.3% of F1 score for Neural Network Model. The result confirms that DL algorithm provides more accurate prediction leading to higher accuracy.

S. Abdullah (✉) · R. Mohamad
Universiti Malaysia Terengganu, Kuala Terengganu, Malaysia
e-mail: syahida.abdullah@umt.edu.my

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_33

1.2 Process Overview

The Malaysian small and medium-sized enterprise (SME) sector has been contributing significantly to the nation's economic development, which accounts 97% (SME Corp, 2021) and 38.2% to national GDP (DOSM, 2020). It also provides employment opportunities for 7.3 million people (DOSM, 2020) in Malaysia, signaling its importance to the Malaysian economy. Despite its economic significance, SMEs have continuously been facing different kinds of risk that include credit risk, operational risk, market risk, legal risk, reputational risk, and cyber risks. This study will look into credit risk as it has been an unsolved issue and the financial institutions (Fis) have been using different kinds of tools to assess the credit risk of the SMEs. In recent years, some FIs have started approaching artificial intelligence (AI) algorithms through machine learning (ML) and deep learning (DL) models to address the issues of credit risk. Basically, ML and DL can help the FIs to classify their customers into different risk segments based on their characteristics and behavior patterns. ML and DL can also help the FIs to detect anomalies or outliers in their data that may indicate potential fraud or online scams. As such, this paper studies comparatively both the Machine Learning and Deep Learning models to provide valuable insights into their practical applicability and potential impact on enhancing sustainable performances of the SMEs.

2 Literature Review

Credit risk is the risk that a borrower will default on their loan, meaning that they will not be able to repay the principal or interest on the loan [3]. It is one of the most important risks that financial institutions face, and it can have a significant impact on their profitability and stability [4]. There are a number of factors that can contribute to credit risk, including the borrower's financial condition, the industry in which they operate, and the overall economic environment [7]. Financial institutions typically use a variety of methods to assess credit risk, including credit scoring models and financial statement analysis [14]. Financial institutions can also use a variety of risk management techniques to mitigate credit risk, such as diversification, collateral, and credit insurance [12].

The non-performing loans (NPLs) in the banking system in Malaysia accounted for 1.4% of total loans at the end of 2022 [13]. There are a number of factors that have contributed to the increase in NPLs in Malaysia in recent years, which has led to a decline in economic activity and an increase in unemployment [10]; the rising cost of living, which has made it more difficult for borrowers to repay their loans; the increasing reliance of businesses and consumers on debt [9].

Hence, risk assessment is a critical aspect of evaluating potential hazards and uncertainties that may affect the performance and stability of businesses, particularly for small and medium-sized enterprises (SMEs) (Saxena et al., 2023). In recent

years, the integration of artificial intelligence (AI), machine learning (ML), and deep learning (DL) models have shown promising potential in enhancing risk assessment processes across various domains (Shi et al., 2022). AI models offer a more data-driven and automated approach, enabling financial institutions to assess credit risk more efficiently and accurately (Shi et al., 2022). Incorporating big data technology into credit risk assessment helps to leverage diverse data sources that lead to more accurate evaluations of SMEs' creditworthiness (Saxena et al., 2023). This data-driven approach can enable financial institutions to make better-informed lending decisions for Malaysian SMEs. In fact, the application of AI, ML, and DL in credit risk assessment offers efficient, data-driven, and automated methods for evaluating credit risk. By addressing challenges related to data imbalance and model transparency, financial institutions can enhance risk assessment processes and support the growth of Malaysian SMEs (Shi et al., 2022).

3 Research Methodology

This research uses publicly available data published by the Central Bank of Malaysia on the statistics of financial data of loan approved, loan disbursed and loan outstanding of the small and medium enterprises (SMEs) companies in Malaysia for the duration from July 2021 until May 2023, the period after the COVID-19 pandemic where the economic activities grew from -5.5% in 2020 to 3% in 2021. The variables used are the amount of loan disbursement to SMEs; amount of loan repayment by SMEs; and amount of outstanding loan for the SMEs. Loan disbursement refers to the amount of loan that has been received by entrepreneurs; loan repayment refers to the amount of loan that has been paid back; and loan outstanding refers to balance of the loan on the part of the entrepreneurs. We have referred to the data from all the sectors that include agriculture, forestry and fishing; mining and quarrying; manufacturing; electricity, gas, steam and air conditioning supply; water supply, sewerage, waste management and remediation activities; construction, wholesale and retail trade; accommodation and food services activities; transportation and storage; information and communication; financial and insurance/ takaful activities; real estate activities; professional, scientific and technical activities; administrative and support service activities; education, health and others; and other sector.

In order to determine the risk or repayment, the data for the difference between loan disbursement and loan repayment, and the difference between loan disbursement and outstanding loan were derived from the dataset and analyzed. The study uses machine learning algorithm for data clustering model to first cluster the data into group of repayment risk. The process was completed using K-Means clustering algorithm.

3.1 Clustering Algorithm

Clustering is an unsupervised machine learning task. Clustering algorithms divide data points and group them into several groups with similar traits and characteristics. The method is used for one of the following purposes: to find an optimal partition to divide the data into specified number of clusters; to find a way to structure a hierarchy; or to find a method based on probability model for cluster modeling [6]. K-means uses an iterative process of assigning each of the data points to groups and cluster them based on similar features. The objective is to minimize the sum of distances between the data points to identify the correct group for each data point.

K-means algorithm is cited as one of the clustering methods which best optimizes the clustering result and redistribute the target set to each clustering center for optimal solution [11, 15]. The dataset used for this study are raw dataset from the Central Bank of Malaysia, thus, the data need to be clustered into categories that shows the ability of repayment of loan for the SME borrowers. This risk is measured based on the difference between loan disbursement and, loan repayment for the SME, and the difference between loan disbursement and loan outstanding amount of the SME. Using the K-mean clustering algorithm enable the development of three clusters showing 3 types of risk among the borrowers in SME companies in Malaysia—the high risk, moderate risk and low risk group.

3.2 Classification Algorithm

In machine learning, classification refers to a predictive modeling problem where a class label is predicted for a given input data. Classification predicts a categorical class label by constructing a model based on training dataset and uses these values to classify new data. It involves a two-steps process of model construction and model usage. Model accuracy which is the percentage that the rest datasets correctly classify the new data is obtained during the model usage steps. The accuracy of the model determines the best model and suitable parameters for classifying future data. The study employed four different methods of classification algorithms and evaluated each model that provides better performance with the available dataset. The models employed were SVM, Decision Tree, Neural Network and Random Forest.

3.2.1 Support Vector Machine

Support vector Machine (SVM) is a machine learning method used to separate data points and find a hyperplane in an N-dimensional space which can classify the data. The goal of SVM model is to generate mathematical functions that will map input variables to desired outputs for classification or regression type prediction problems. Thus, SVM uses nonlinear kernel functions to transform non-linear relationships

among the variables into linearly separable feature spaces. This will transform the data into its required form. The mostly used kernel function is the radial basis function (RBF) which will also be employed in our model.

SVM will then construct the maximum-margin hyperplanes to optimally separate different classes from each other based on the training dataset. A hyperplane is a geometric concept used to describe the separation surface between different classes of data. In SVM, two parallel hyperplanes are constructed on each side of the separation space with the aim of maximizing the distance between data. We employ SVM model with RBF kernal function and 100 iteration limits.

3.2.2 Decision Tree

The decision tree process starts with constructing a top-down recursive tree structure with a divide-and-conquer manner, where the dataset is partitioned recursively based on selected attributes. The root node at the top is a point that contains the starting dataset. Each subsequent classification decision is called a decision node and a class label is represented by a leaf node which becomes the final classification for the data. The branches represent the test results. The second part of the decision tree process is the tree pruning, to identify and remove branches that reflect noise or outliers in the dataset. The most common splitting function used in decision tree classification algorithms are GINI Index or Information Gain criteria.

A decision tree can lead to overfitting where too many branches were generated thus affecting the accuracy of the model. One approach used to avoid overfitting is by performing the tree pruning, either by halting the tree construction, or by removing the branches from the decision tree.

3.2.3 Random Forest

A random forest is essentially a collection of decision trees, where each tree is slightly different from the others. Like decision trees, Random Forest are also used for classification purposes, however, one of the main problems with decision trees is that they tend to overfit the training data and thus random forest attempt to address this problem.

The idea behind random forests is that each tree might do a relatively good classification job, but will likely be overfit on part of the data. Thus, if we build many trees, all of which work well and overfit in different ways, we can reduce the amount of overfitting by aggregating their results. The reduction in overfitting, while retaining the classification power of the trees, can be shown using rigorous mathematics functions and statistics measures in random forest. RFs ensure that each tree is different by either selecting the features in each split, or by selecting the data points used to build a tree.

Random forest algorithm method is called ensemble methods, where a predictor ensemble is built with several decision trees that expand in randomly selected data

subspaces [2], thus creating a more powerful tree. Random forest models are known to have produced an accurate classification result and are efficiently used for large databases.

3.2.4 Neural Network

Neural networks consist of a stack of one or more elementary sub-structures, called layers, each one of which is composed of one or more nodes. In general, the structure of a neural network includes: an input layer, which receives as input the model independent variables; an output layer, which produces the final result of the model prediction; and, one or more hidden layers, with each one of which receives the input from either the input or the immediately preceding hidden layer, and sends the output to either the output or the immediately following hidden layer [8]. This multiple layer of nodes combines all variables together in a highly non-linear way, and, an optimization procedure is used to let the Network learn the input–output dependency directly from data.

Neural network is a model categorized as ensembles method that combines multiple machine learning models to create more powerful models. The combination of these several models enables the model to obtain better generalization performance and accuracy [5].

4 Result and Discussion

In K-Mean clustering algorithm, silhouette values are used to determine the optimal number of clusters when performing K-Means algorithm thus measuring the quality of the clustering. Silhouette values are a coefficient between -1 and 1, with score 1 denotes the best values of which the data point is very compact within its cluster, and far from other clusters. A value near 0 represents overlapping cluster, and a negative cluster indicate that samples might have been assigned to wrong clusters.

Using K-Mean clustering algorithm, our model best fits into three clusters of risks—Low Risk, Moderate Risk and High Risk, with low risk showing the SMEs with highest capability to repay their loan, and high risk showing the possibility that the SME would not be able to repay their loan. Analysis of the silhouette values of the clusters produced the result as shown in Fig. 1. The silhouette values for the model range from 0.5 0.53 for C2 (Low Risk), 0.53 to 0.645 for C3 (Moderate Risk) and 0.645 to 0.72 for C1 (High Risk).

The result of the clustering model then is supplied to the classification model. We employed four different types of prediction models: Support Vector Machines, Decision Tree, Random Forest and Artificial Neural Networks. The pre-processed data is sent to the classification model. Each model is trained and calibrated using the appropriate parameters. Figure 2 illustrates the process for model development where we test and calculate the accuracy of each model.

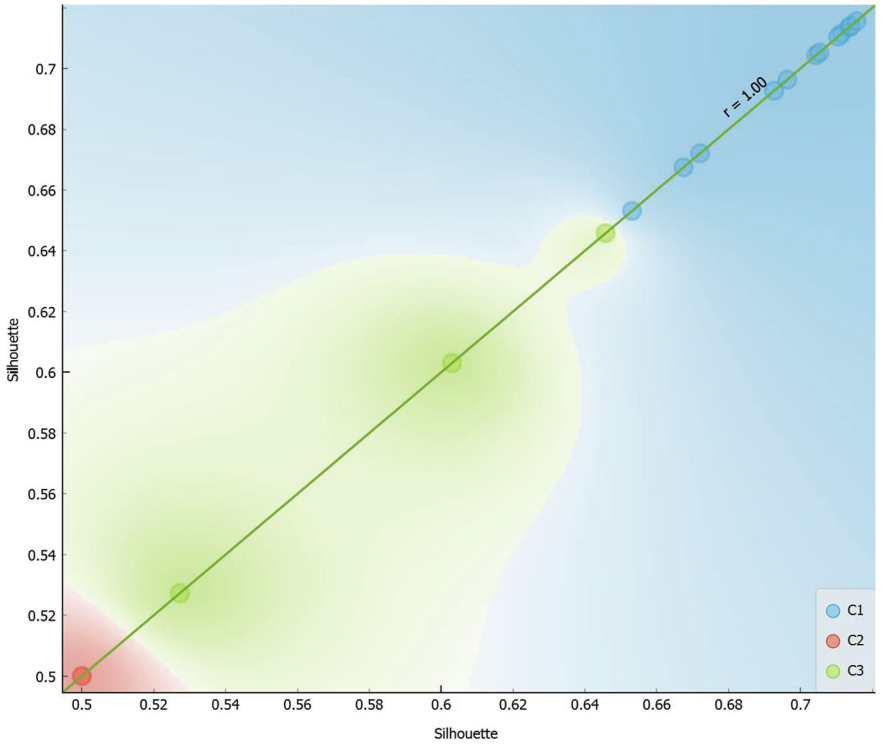


Fig. 1 Scatter Plot of the cluster obtained from K-Mean Clustering Algorithm showing 3 clusters of Risk

Machine learning prediction is a result of assorted processes such as data preparation, choosing appropriate model, train the chosen model, and finally validating the model. Only after the validation process is completed, the model is efficient to make perform prediction. In model validation process the trained model is evaluated with testing data set, which could either come from the same dataset (in-sample dataset), or from another out-of-sample dataset. Our model user uses a random sampling validation procedure to randomly split the data into the training and testing set of the proportion of 70:30. These results are used to determine the best classification model to use for the dataset.

For measuring the performance of the four classification modes, the following scores are analyzed:

- AUC or Area Under the Curve represents the degree or measure of separability which tells how good the model is capable of distinguishing between classes. The higher the AUC, the better the model is at predicting the classes;
- CA or Classification Accuracy refers to the proportion of correctly classified examples in the data set;

- Precision score shows the ability of the model classifying data to a relevant data point. It is a proportion where all positive predictions were correct;
- Recall score shows the proportion of the correct positive prediction out of all positive predictions made by the model. This score provides indication of missed positive prediction by the model;
- F1 score shows the accuracy of the model by finding the balance between precision and recall by calculating their harmonic mean. It is a measure of a test's accuracy where the highest possible value is 1 which indicates perfect precision and recall; and
- Specificity score which shows the proportion of trues negatives among all the negative instances.

Table 1 shows the confusion matrix of the predicted proportion for the four models. The result shows that the prediction made by Random Forest model predicts 88% of data correctly for C1 (High Risk) group, and 89.2% for Neural Network model accordingly (for C1 group). Similarly, the prediction for C3 group for both Random Forest model and Neural Network model were 82.4% and 76.7% respectively. The result also shows that no data from the sampling are grouped in C2.

In comparison to Support Vector Machine and Decision Tree model, the prediction proportion obtained were slightly lower: the C1 predicted proportion were 84.8% and 75% respectively for Support Vector Machine and Decision Tree, while the C3 predicted proportion were 81.4% and N/A for the Decision Tree.

Table 2 summarizes the test results of the four different types of prediction models used. The result shows that the performance accuracy of Random Forest

Table 1 Confusion Matrix for the developed model (predicted proportion)

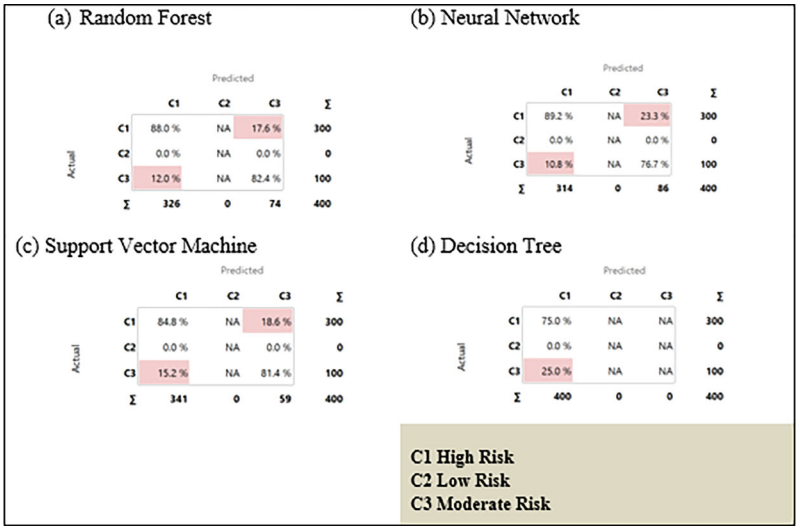


Table 2 Prediction Accuracy Results for all four classification models

Classification Model	AUC	CA	F1	Precision	Recall	Specificity
Random Forest	0.960	0.865	0.861	0.861	0.865	0.728
Neural Network	0.912	0.853	0.843	0.847	0.853	0.651
Support Vector Machine	0.637	0.843	0.827	0.839	0.843	0.601
Decision Tree	0.500	0.750	0.643	0.562	0.750	0.250

and Neural Network model outperformed SVM and Decision Tree Model in all accuracy measures.

From all types of analysis, the Decision Tree is observed to have the least accuracy performance. This result can be evidenced from the Confusion Matrix, and Prediction Accuracy analysis.

4.1 Machine Learning Versus Deep Learning Model for Data Prediction

Machine Learning describes the capacity of systems to learn from problem-specific training data to automate the process of analytical model building and solve associated tasks, while Deep learning is a machine learning concept based on artificial neural networks [5]. For many applications, deep learning models outperform machine learning models in data analysis approaches. Random forest model is essentially a collection of decision trees, where each tree is slightly different from the others, while Neural Network consists of stacks of many elementary sub-structures, called layers with each one composed of one or more nodes. These models contribute to better performance and accuracy of the developed models.

This paper uses two deep learning model models: random forest and neural network. As the result shows, the two models outperformed the machine learning model (support vector machine and decision tree) with classification accuracy of 86.5% for Random Forest and 85.3% for Neural Network. The accuracy for the machine learning model used is 84.5% for Support vector Machine Model and 75% for Decision Tree. Similarly, the F1 score shows the same manner of performance—86.1% for Random Forest, 84.3% for Neural Network (both are deep leaning models), and 82.7% for Support Vector Machine and 64.3% for Decision Tree (where both are machine learning models).

5 Conclusion

The study conducted in this paper demonstrates the power of machine learning tools in predicting risk of loan repayment among SME companies in Malaysia. This is demonstrated using four different prediction algorithms—Random Forest (RF), Neural Network (NN), Support Vector Machine (SVM) and Decision Tree (DT), in which RF and NN are deep learning methods, while SVM and DT are machine learning methods. The results show that the deep learning model produced better prediction result with 86.5% classification accuracy, 86.1% of F1 score for Random Forest Model; and 85.3% of classification accuracy and, 84.3% of F1 score for Neural Network Model.

The result confirms that deep learning algorithm is a powerful tool in loan repayment prediction which leads to high accuracy of the prediction result. Various data analytics tools are available in the market and are proven to be powerful that can be use develop such system in order to promote the digitalization of the loan repayment prediction system in Malaysian financial institution.

References

1. Abdullah, S., Othman, Z., Mohamad, R.: Predicting the Risk of SME Loan Repayment using AI Technology-Machine Learning Techniques: A Perspective of Malaysian Financing Institutions. J. Adv. Res. Appl. Sci. Eng. Technol. **31**(2), 320–326 (2023). https://doi.org/10.37934/araset.31.2.320326
2. Addo, P., Guegan, D., Hassani, B.: Credit risk analysis using machine and deep learning models. Risks **6**(2), 38 (2018). https://doi.org/10.3390/risks6020038
3. Baaquie, B.E., Karim, M.M.: Pricing risky corporate bonds: An empirical study. J. Futur. Mark. **43**(1), 90–121 (2023)
4. Doumpos, M., Lemonakis, C., Niklis, D., Zopounidis, C., Doumpos, M., Lemonakis, C., ... Zopounidis, C.: Introduction to credit risk modeling and assessment. Anal. Tech. Assess. Credit. Risk: Overv. Methodol. Appl. 1–21 (2019)
5. Janiesch, C., Zschech, P., Heinrich, K.: Machine learning and deep learning. Electronic Markets, 31. Springer (2021). https://doi.org/10.1007/s12525-021-00475-2
6. Jothi, R., Mohanty, S.K., Ojha, A.: DK-means: a deterministic K-means clustering algorithm for gene expression analysis. Pattern Anal. Appl. **22**(2), 649–667 (2019). https://doi.org/10.1007/s10044-017-0673-0
7. Kim, J.B., Song, B.Y., Wang, Z.: Special purpose entities and bank loan contracting. J. Bank. Finance **74**, 133–152 (2017)
8. Luca Sitzia, Baccaglini, R., Vittorio Malacchia, Cozzi, F. A Neural Network Approach for the Estimation of Mortgage Prepayment Rates (2021). https://doi.org/10.2139/ssrn.4179429
9. Mahdzan, N.S.A., Abd Sukor, M.E., Ismail, I., Rahman, M.: Consumer Financial Vulnerabilities in Malaysia: Issues. Routledge, Trends and Psychological Aspects (2020)
10. Mahyoub, M., Said, R.M.: Factors Influencing Non-Performing Loans: Empirical Evidence from Commercial Banks in Malaysia. Res. J. Bus. Manag. **8**(3), 160–166 (2021)
11. Shakeel, P.M., Baskar, S., Dhulipala, V.R.S., Jaber, M.M. Cloud based framework for diagnosis of diabetes mellitus using K-means clustering. Health Inf. Sci. Syst. 6(1) (2018). https://doi.org/10.1007/s13755-018-0054-0
12. Supervision, B.: Basel committee on banking supervision. Principles for Sound Liquidity Risk Management and Supervision (September 2008) (2011)

13. Theong, M.J., Lau, W.Y., Osman, A.F.: Comparative study of determinants of the Malaysian household nonperforming loans: Evidence from Nardl. Singap. Econ. Rev. 1–19 (2022)
14. Thomas, L., Crook, J., Edelman, D. Credit scoring and its applications. Soc. Ind. Appl. Math. (2017)
15. Zhu, Z., Liu, N.: Early warning of financial risk based on K-Means clustering algorithm. Complexity **2021**, 1–12 (2021). https://doi.org/10.1155/2021/5571683

Malware Classification in Cloud Computing Using Transfer Learning

Meryem EC-Sabery, Adil Ben Abbou, Abdelali Boushaba, Fatiha Mrabti, and Rachid Ben Abbou

Abstract The adoption of cloud computing has revolutionized the way organizations handle their data and applications. However, this paradigm shift, with its shared infrastructure and remote accessibility, has also introduced big security concerns. One of the most important challenges is the growing threat of malware that may consume CPU, memory, and bandwidth of cloud resources. Consequently, there is an urgent need for malware detection and classification system. In this paper, we propose to use Convolutional Neural Network (CNN) based on three popular fine-tuning techniques to classify binary files of malwares in cloud computing. The experiments are applied on Malimg dataset, which contains grayscale images of 25 families of 9,339 malwares and ResNet50 transfer learning model reached a good accuracy with 98.29%.

Keywords Cloud · Convolutional neural network · Malware image · Transfer learning

1 Introduction

Cloud computing [1], refers to a network of remote and shared servers over the internet. These resources are made available to users and organizations for purposes such as data storage, processing, and software access, with the key attributes of on-demand self-service, scalability, and measured usage known by pay-as-you-go. The "pay-as-you-go" technique, often used in cloud, can be misused by cybercriminals by sending malicious attachments or links to the cloud user and exploit his resources to host and distribute malwares [2]. This puts the cloud user's budget, reputation and security at risk.

Malware encompasses a variety of malicious software, including viruses, worms, and Trojans [3, 4] . The lifecycle of a virus comprises dormant, propagation, trigger, and execution phases. During execution phase, it initiates background processes that

M. EC-Sabery (✉) · A. Ben Abbou · A. Boushaba · F. Mrabti · R. Ben Abbou
Intelligent Systems and Applications Laboratory, Faculty of Sciences and Technology Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: meryem.ecsabery@usmba.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_34

consume CPU cycles, potentially modifying or deleting files, affecting system performance and intercepting or stealing sensitive information on the infected system. Trojan is malware that disguises itself as a legitimate or benign program to deceive users and gain unauthorized access to their computer systems. The main objective of worms, is to replicate and spread themselves as widely as possible within computer networks, depleting system resources like storage and bandwidth. Continuous monitoring of system parameters, such as CPU, memory, and network data, provides real-time input to deep learning models for malware detection.

Traditional approaches for malware detection rely on signature-based and heuristic methods [5]. In signature-based detection, algorithms or hash values are used to identify malicious software. This method looks for similarity between the signature stored in its database and the signature generated for the unknown file. Thus, a simple modification in software code generates a new signature making this method ineffective in identifying malware that has undergone a code change. In heuristic methods detection, the experts must analyze the behavior of malware to establish a set of detection rules. The drawback of this approach is that it requires preliminary malware analysis, which is impossible given the increasing number of malware in cloud environments and so this method fails in detecting unknown and new malware. An alternative approach to identify malicious code involves the use of static and dynamic analysis methods [3, 6]. In static analysis, the code or structure of the program is examined without the execution of potentially malicious code, if encrypted or compressed segments are added to the program codes, static methods cannot detect the malware. In dynamic analysis methods, the program is executed within a controlled environment as a sandbox, and all its actions and interactions with the system and external resources are then monitored. However, dynamic approaches require a large amount of computational resources to detect malware, and there is a possibility that some malicious behavior goes unnoticed because the environment does not satisfy the triggering conditions.

Convolutional Neural Network (CNN) based malware detection has gained popularity due to its ability to handle complex data representations and learn complex patterns that may be challenging for traditional signature or heuristic methods especially in cloud computing, which has seen a dramatic growth of malwares. References [7–9] presented a CNN model, which is based on monitoring the process performance during malware execution in cloud environments. This model is trained on dataset containing images of the process performance metrics, i.e. memory, cpu, input/output networks. In [10, 11], CNN is trained on dataset that contains grayscale image of portable executable of several malwares. Image-based analysis by deep learning techniques can be a valuable approach for understanding malware patterns, because of its ability to understand large data and to extract meaning from new data without the need for a cybersecurity expert knowledge.

In this work, we proposed to use pretrained CNN models to classify malware images in cloud computing given their speed during training, saving time and computational resources compared to using CNNs from scratch. The experiment conducted with the help of Malimg dataset and at the end, an evaluation of results is performed to select the best CNN model for malware classification.

The remainder of this paper is structured as follows: Sect. 2 briefly presents some related work of malware detection and classification in cloud computing, Sect. 3 introduces the CNN model, Sect. 4 presents the proposed method for malware classification in the cloud, Sect. 5 offers performance evaluation of the used CNN models and we conclude in Sect. 6.

2 Related Works

Deep neural networks, especially CNNs have caught the attention of several researchers for detecting malware in cloud environments. The following are some stat of arts about image classification by training CNN on visual representations of malwares or their behavior.

In [7], authors proposed a malware detection approach in cloud computing using CNN. The experiment was conducted on openstack to simulate a real world scenario, 25 malware binaries belong to rootkits, trojans and backdoors are randomly obtained from VirusTotal and injected one per experiment to ubuntu VM in openstack and performance metrics are collected. In this works authors selected 120 processes to monitor, for each process 28 features are measured for example cpu usage, network information, memory information, etc.., The proposed CNN takes fixed-size images m*n as inputs, where n the number of features and m the number of processes. 2d CNN model reaches an accuracy of 79% and 3d CNN model significantly improves the accuracy to 90%.

In [8], authors proposed an online malware detection in cloud IaaS, they prepared the same test environment as [7], 113 malware binaries are injected to VM and process performance metrics are collected. The malware image detection is performed based on various CNNs, which are LeNet, DenseNets and ResNets. The accuracy reachs 93%. Authors in [9] injected 40,680 malicious and benign samples on cloud VM, and then collected performance metrics. They applied for an online malware detection different baseline machine learning models including, support vector classifier (SVC), random forest Classifier (RF), CNN, etc. CNN has given the best overall performance.

In [12], authors proposed CNN based malware detection in cloud environments, they used for training a dataset which contains system metrics, authors used SMOTE technique to generate new data in order to reduce the imbalance problem. The dataset is splitted into three files CPU data, memory data and network data and these files are transmitted as an input to the CNN model. CNN and Lenet-5 models are used and reached an accuracy of 95%.

In [10], authors suggested and compared several CNN architecture based on local response and batch normalization for malware classification. They evaluated these models by training them on Malimg dataset, which contains grayscale image of 25 families of malware binary files, They experimented a various size of these images but the largest 128x128 outperformed the rest with 95,07% of accuracy, however the training time remains very long compared to the small size of images used for training.

In [11] authors proposed an approach based on CNN for detecting new and unknown malware, The model includes two phases, training and detection. For the malware recognition, binary codes are converted into images and then trained by the proposed CNN model to determine its nature. For the training phase, CNN is tuned on Malimg and on benign software datasets.

3 CNN Architecture

Deep learning has seen remarkable advancements in recent years, driven by various factors including increased computational power and innovations in neural network architectures [13]. CNNs models are highly effective at finding significant patterns in visual data by techniques as convolution, pooling, and other computations, without the need to have a strong knowledge in the area studied. Researchers in the field of cybersecurity and malware analysis have also taken advantage of CNNs for malware detection and classification. They treated malware as image, either by converting its PE binary file or the performance metrics while it's running, i.e., memory data, cpu data, network data, system calls, etc. into images. CNNs extracted the important malware characteristics from these images and performed high accuracy of detection. In this work, we have implemented three CNN models by using transfer learning technique for malware images classification.

CNN [14] is a type of artificial neural network specialized in processing and analyzing visual data, particularly images. It is designed to learn and identify spatial hierarchies of features from the input data. CNNs are mainly used in computer vision tasks, such as object detection, image classification and more. Figure 1 shows the CNN architecture, it includes the following layers:

- Convolutional layers apply convolution operations to the input data, using learnable filters to detect features like edges, textures, or more complex patterns.
- Pooling layers down-sample the spatial dimensions of the feature maps, reducing computational complexity while preserving important features.

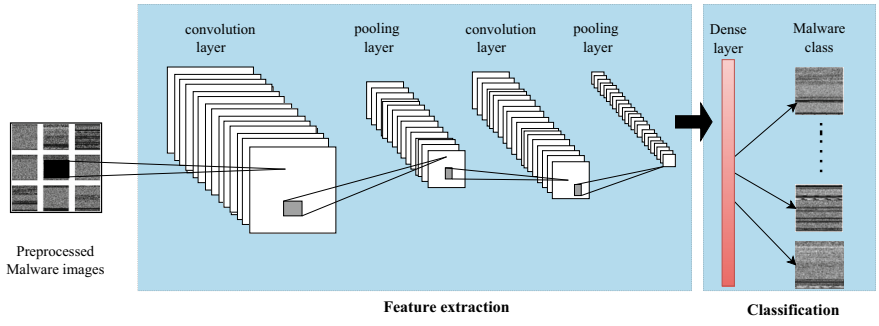


Fig. 1 CNN architecture

- Connected layers (dense layer) link the high-level features from the previous layers to the output layer for classification tasks.

CNNs use backpropagation to adjust the weights and biases of the network in order to minimize the loss function. They often require large labeled datasets for an effective training.

4 The Proposed System for Malware Classification in Cloud Computing

The flowchart in Fig. 2 presents the proposed system of malware classification in cloud computing. It includes three modules: data preprocessing, CNN training and malware classification.

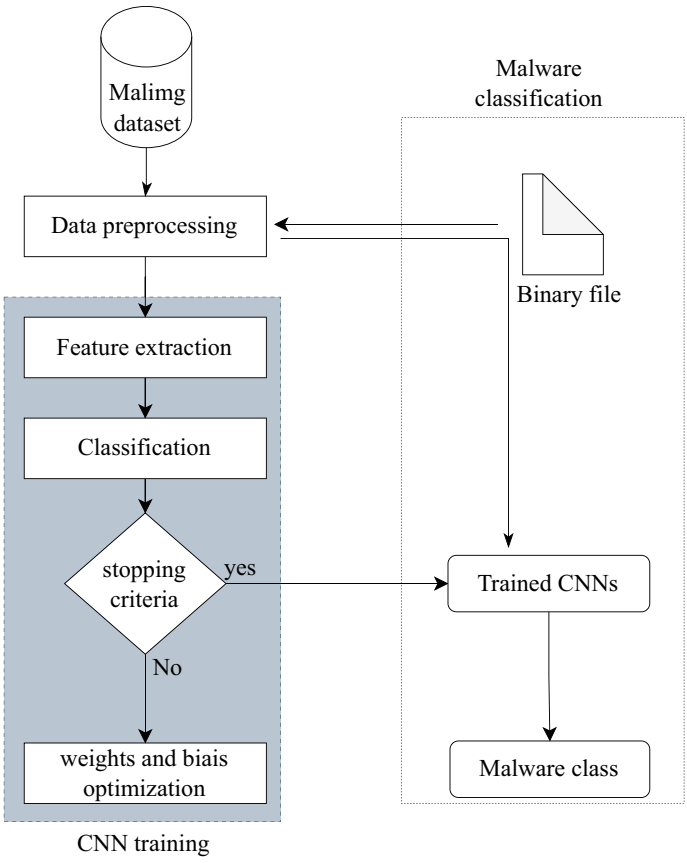


Fig. 2 The flowchart of the proposed malware classification system

1. Data preprocessing involves that the input data is in a suitable format for training the CNN, it is about collecting binary file, replacing every 8 bit by its decimal value, reshaping the result to 2D matrix, which is interpreted as image, resizing it to have the same dimensions of trained images. Normalizing it, by scaling pixel values to the common range [0,1]. The output of this module is utilized as input for a CNN network for training and for malware classification.
2. CNN training comprises two big components for image classification, which are features extraction and classification.

   - Feature extraction takes place first in CNN networks to capture relevant features in the input data, It is typically composed of convolutional and pooling layers. Convolutional layers detect features like edges and textures in the input data using small learnable filters. Pooling layers are typically placed after convolutional layers and serve to reduce the spatial dimensions of the feature maps while retaining important information and making the network computationally efficient, the most popular pooling method are max pooling and average pooling. The output of this process feed the CNN classification part.
   - Classification: refers to the process responsible for making predictions. The feature maps from the previous layers are flattened into a one-dimensional vector and serve as input to the dense layers. The final layer of the dense layers often employs a softmax activation function to provide class probabilities, allowing the CNN model to classify the image into one of the predefined categories.

   In the training phase of CNN model, the model's weights and biases are adjusted by using an optimizer to minimize the loss function and maximize the accuracy rate. The accuracy metric in Eq. (1) is calculated based on the following parameters: True positive (TP) and true negative (TN): Number of images accurately classified. positive (FP) and False negative (FN): Number of images incorrectly classified.

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

3. Malware classification: At this stage, the image prepared by the preprocessing module is passed to the trained CNN model, the output is calculated and then the class, which has the highest probability, is affected to the program.

5 Experiments and Results

In this experiment, we used a machine with i5 CPU 1.70 GHz, 16 Go of RAM and Keras library to import pretrained CNN models. We have no graphical processing unit, so the training process for each of these models has taken a long time, approximately three hours.

5.1 Dataset Description

All models developed in this work are trained and tested on Malimg dataset, it contains 9,339 images of malware binaries belonging to 25 families. Specifically, the content of malware binary files are converted into grayscale images. In Malimg the number of malware from each family is not equal, Allaple.A malware family represents the most numerous with 2,949 samples. Figure 3 shows some examples of malwares from Malimg dataset.

5.2 Building CNN Models Using Transfer Learning

In our work, Three CNN models based on transfer learning techniques from the stat of arts are applied for malware classification. Due to our limited computing environment and the small malware dataset, we used instead of training a CNN from scratch, a pretrained models, these models are trained on Imagenet dataset. We chose to keep the optimized weights of all layers of these models, we added a flatten layer

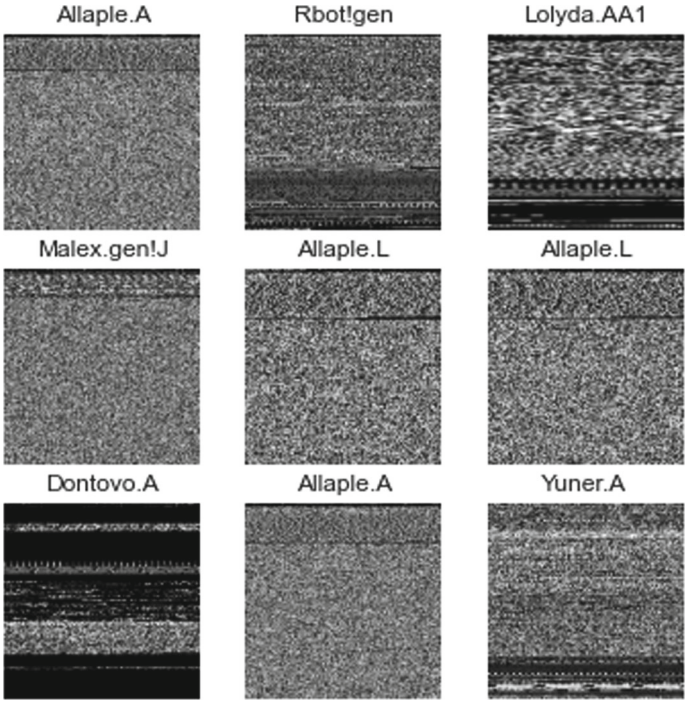


Fig. 3 Sample of Malimg Dataset

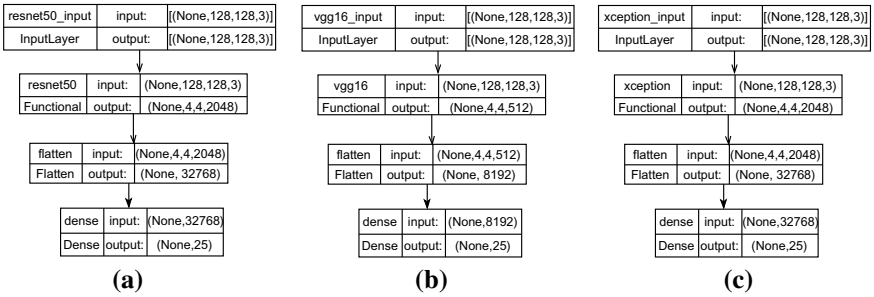


Fig. 4 Architecture of the proposed transfer learning model : **a**) ResNet50, **b**) Vgg16 and **c**) Xception

and one dense layer with 25 neurons that represent the number of malware families (only the fully connected layer weights are trained). Softmax activation is used in the output layer and during the model compilation Adam optimizer is also used. In order to prevent overfitting, an early stopping is configured, if model's performance on the validation degrades within three successive iterations, training may be stopped. Figure 4 shows the architecture of the selected CNNs models.

5.3 Validation and Testing the Given CNN Models

CNN training, validation, and testing are distinct phases in the development and evaluation of our proposed models. validation set for tuning the models hyperparameters during epochs and avoiding overfitting, and the testing set provides a final measure of the model's effectiveness. Table 1 depicts the data used for these phases: In this step, we can get the performance and evaluation metrics for our models before trust them for real case classification. Figure 5 presents the evolution of malware classification accuracy during epochs by our CNN model based on the most popular pre-trained CNN models VGG16, Res-Net50 and Xception. The three models performed good for fine-tuning task, Fig. 5a showed that ResNet50 accuracy has seen a quick improvement until the 6 epoch and then the rhythm of evo-lution became

Table 1 Training, validation and testing set

	Number of malware images
Training	5282
Validation	1494
Testing	2563

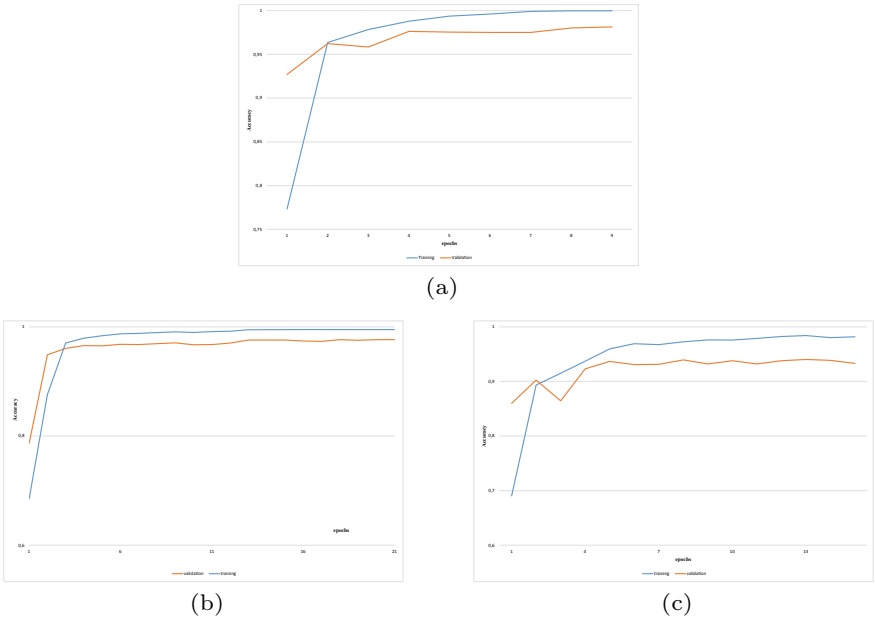


Fig. 5 Accuracy curve for : ResNet50 (**a**), Vgg16 (**b**) and Xception (**c**)

slow without losing the convergence. This model achieved after 16 epochs the highest value of validation accuracy with 98.29%. The convergence between training and validation accuracy is excellent, making this model the most suitable choice for malware classification.

VGG16 model in Fig. 5b showed a good accuracy convergence, it reached its maximum value of accuracy 96% after 8 epochs and then after three epochs the model stopped to prevent the overfitting.

For Xception model, the results are not good enough compared to VGG16 and ResNet50, the accuracy improvement stopped from the 5 epoch and the space between training and validation still large and thus the performance of the model not suitable for our classification.

Loss curve serves also as a valuable tool for assessing models. In our work, Fig. 6 presents loss function evolution within epochs for VGG16, ResNet 50 and Xception models. Similarly, to accuracy curve, this tool confirms that ResNet50 model outperformed the other models in classifying malwares, and Xcpetion model gave a poor result shown by the significant gap between the training loss curve and validation loss curve.

Regarding confusion matrix in Fig. 7, ResNet50 and VGG16 provided close testing classification results except for these malware families : Swizzor.gen!E and Swizzor.gen!I. ResNet50 model well predicted the Swizzor.gen!E family by 73.17% while VGG16 failed in 34% cases. For Swizzor.gen!I family, ResNet50

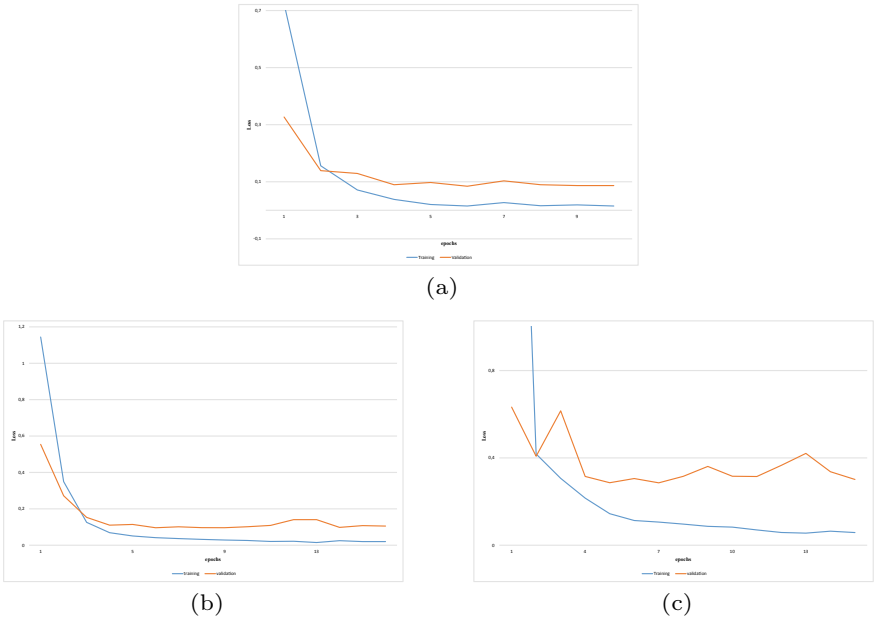


Fig. 6 Loss curve for : **a**) ResNet50, **b**) Vgg16 and **c**) Xception

well predicted the malware family by 56.25% while VGG16 model identified 48% of this malware family. In general, ResNet50 model obtained results with acceptable advantage of VGG16. Xception in Fig. 7c showed that for the most malware families, the model is less efficient in predicting its correct family compared to ResNet50 and VGG16, it totally failed at identifying malware of Autorun.K family.

ResNet may be a good tool to malware classification. Time and effort for cybersecurity experts in malware detection can be saved and the accuracy of classification and detection may increase making the cloud environments more secure.

The critical drawback of CNNs architectures is they can be vulnerable to adversarial attacks. Adversarial attacks involve intentionally crafting input images to mislead a neural network. These crafted inputs are often imperceptible to humans but can cause CNN models to make incorrect predictions; considering a malware as a normal program file posing significant security risks and potential disruptions to cloud services. To enhance the model's resistance against adversarial attacks, researchers engaged in understanding their influence on image classification and they are developing robust model architectures rely on adversarial training.

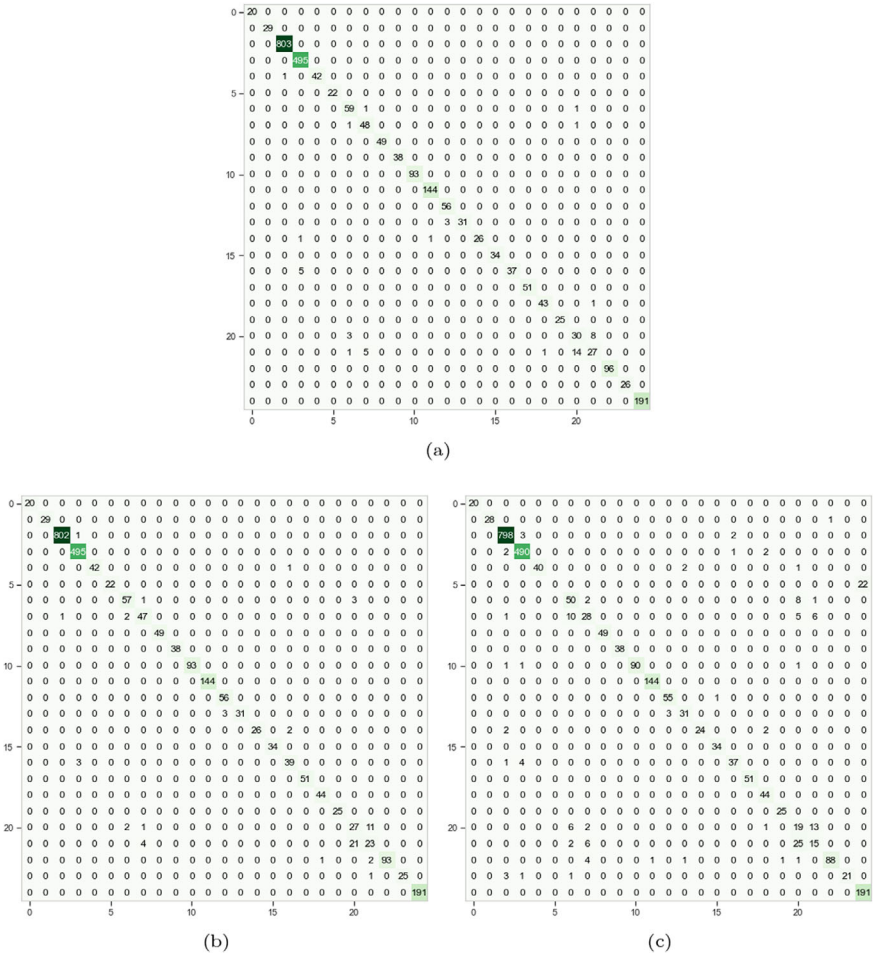


Fig. 7 Confusion matrix of : **a**) ResNet50, **b**) Vgg16 and **c**) Xception

6 Conclusion

In this work, we proposed a CNN based on fine-tuning techniques, taking the most popular pretrained model in the literature and adapting them to classify malware images. The performance of ResNet50, VGG16 and Xception were evaluated on Malimg Dataset. The proposed models did not require any features engineering or a big knowledge on cybersecurity field. This makes the method more flexible and effective for detecting malware in cloud computing, given its constantly exposed nature to multiple threats and many new and unknown malwares. The experiments demonstrated that the ResNet50 model achieved a validation accuracy of 98.29%, which was the highest, followed by VGG16. Thus, converting malware into an image

can yield favorable results, we have just to use images with good quality and tune the used algorithms.

References

1. Rashid, A., Chaturvedi, A.: Cloud computing characteristics and services: a brief review. Int. J. Comput. Sci. Eng. **7**(2), 421–426 (2019)
2. Nadeem, M.A.: Cloud computing: security issues and challenges. J. Wirel. Commun. **1**(1), 10–15 (2016)
3. Aslan, Ö., Ozkan-Okay, M., Gupta, D.: A review of cloud-based malware detection system: opportunities, advances and challenges. Europ. J. Eng. Technol. Res. **6**(3), 1–8 (2021)
4. Chandy, J.: Review on malware, types, and its analysis. IJRASET **10**, 386–390 (2022)
5. Aslan, Ö.A., Samet, R.: A comprehensive review on malware detection approaches. IEEE Access **8**, 6249–6271 (2020)
6. Shafin, S.S., Karmakar, G., Mareels, I.: Obfuscated memory malware detection in resource-constrained iot devices for smart city applications. Sensors **23**(11), 5348 (2023)
7. Abdelsalam, M., Krishnan, R., Huang, Y., Sandhu, R.: Malware detection in cloud infrastructures using convolutional neural networks. In: 2018 IEEE 11th International Conference on Cloud Computing (CLOUD), pp. 162–169. IEEE (2018)
8. McDole, A., Abdelsalam, M., Gupta, M., Mittal, S.: Analyzing cnn based behavioural malware detection techniques on cloud iaas. In: Cloud Computing–CLOUD 2020: 13th International Conference, Held as Part of the Services Conference Federation, SCF 2020, Honolulu, HI, USA, September 18–20, 2020, Proceedings 13, pp. 64–79. Springer (2020)
9. Kimmell, J.C., Abdelsalam, M., Gupta, M.: Analyzing machine learning approaches for online malware detection in cloud. In: 2021 IEEE International Conference on Smart Computing (SMARTCOMP), pp. 189–196. IEEE (2021)
10. Kiger, J., Ho, S.-S., Heydari, V.: Malware binary image classification using convolutional neural networks. In: International Conference on Cyber Warfare and Security, vol. 17, pp. 469–478 (2022). Academic Conferences International Limited
11. Kumar, R., Xiaosong, Z., Khan, R.U., Ahad, I., Kumar, J.: Malicious code detection based on image processing using deep learning. In: Proceedings of the 2018 International Conference on Computing and Artificial Intelligence, pp. 81–85 (2018)
12. Kotian, P., Sonkusare, R.: Detection of malware in cloud environment using deep neural network. In: 2021 6th International Conference for Convergence in Technology (I2CT), pp. 1–5. IEEE (2021)
13. Moujahid, H., Cherradi, B., Gannour, O.E., Bahatti, L., Terrada, O., Hamida, S.: Convolutional neural network based classification of patients with pneumonia using x-ray lung images. Adv. Sci. Technol. Eng. Syst. J. **5**(5), 167–175 (2020)
14. Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M.A., Al-Amidie, M., Farhan, L.: Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. J. big Data **8**, 1–74 (2021)

An Assessment System for ML-Based XSS Attack Detection Models Between Accuracy Coverage and Data

Maryam Et-tolba, Charifa Hanin, and Abdelhamid Belmekki

Abstract The huge capability of ML-based solutions has impacted various areas including cybersecurity. Those solutions can detect and respond to several threats by offering advantages over traditional rule-based methods. A serious web security threat is Cross-Site Scripting (XSS) attack, it occurs when an attacker inserts malicious content into a vulnerable web application to perform unauthorized actions. Out of the various XSS attack detection techniques and methods, ML-based models have a profound ability to achieve impressive results and improved accuracy. However, an evaluation and validation of those approaches is required. In this study, we present an assessment system for pre-existing ML-based XSS attack detection models. Our assessment system contains two phases, the first phase aims to understand the model's architecture, and the second phase ensures the model's efficiency across unseen datasets. In experiments, we applied our evaluation system to an existing model, results highlight some limitations such as integrity, interpretability, overfitting, and false positive rates.

Keywords Cybersecurity · XSS attack detection · ML models · Assessment system

1 Introduction

Nowadays, the large spread usage of web applications in critical fields attracts malicious actors who seek valuable data and content. They exploit vulnerabilities in dynamic real-time web interaction between web client and application server. We

M. Et-tolba (✉) · C. Hanin · A. Belmekki
INPT (Institut National Des Postes Et Télécommunications), Rabat, Morocco
e-mail: ettolba.maryam@inpt.ac.ma

C. Hanin
e-mail: hanin@inpt.ac.ma

A. Belmekki
e-mail: belmekki@inpt.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_35

can also admit that many efforts are made by Internet community to strangling web application against attack, mitigate and reduce the risk.

Among the most prevalent targeting web applications is Cross-Site Scripting (XSS). This last allows an attacker to exploit vulnerabilities in order to insert malicious code content into a website (Fig. 1). The lack of suitable measurements and efficient protection referred typically to leave critical vulnerabilities allowing attackers to easily succeed their attack. Different types of data can be relieved by running an XSS attack, from the user (e.g., cookies, session tokens, login credentials, etc.) and/or similarly from the browser (e.g., operating system, plugins, versions, etc.). Moreover, several attacks such as phishing, Distributed Denial of Service (DDoS), Cross-site Request Forgery (CSRF), and Remote Code Execution (RCE) can be enabled directly or indirectly due to XSS. On its own, the XSS menace is already a major challenge, but if combined with other attack techniques it can duplicate extra damages.

More than that, the XSS attack is becoming increasingly more sophisticated. Certain limitations of traditional mechanisms allow attackers to bypass easily access controls by encoding or obfuscating their malicious payloads. Moreover, when it comes to the quality of alerts generated by different countermeasure mechanisms, and despite the robust defense mechanisms that have been made and designed by researchers and cybersecurity professionals to counter the escalating menace of XSS attacks, we can obviously note that these mechanisms may generate an important rate of false positive and false negative which in the end confirm that XSS attack remains a big challenge. In the same way of strangling web applications against XSS attacks, various promising approaches integrate Machine Learning (ML), Deep Learning

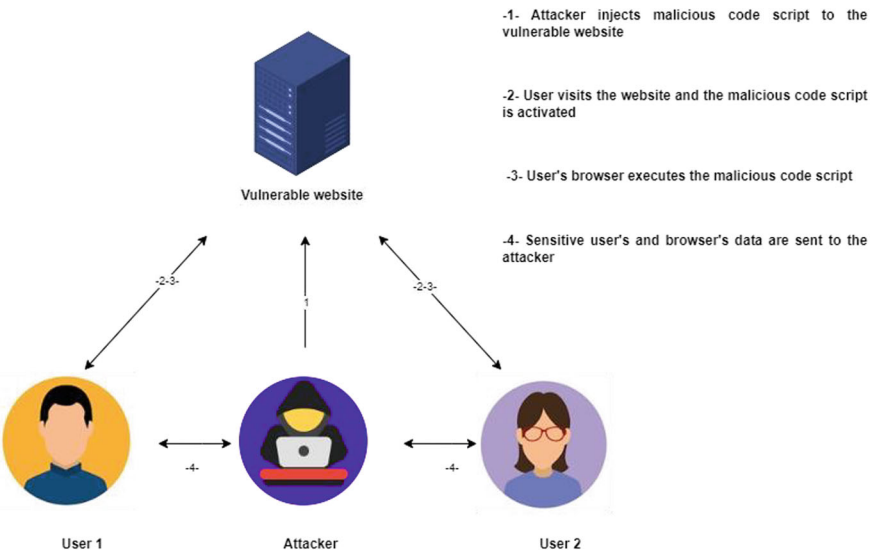


Fig. 1 Cross-Site Scripting attack scenario

(DL), and other advanced algorithms in this context to deal with this challenge. ML models possess the inherent capability to recognize patterns, discern anomalies, and adapt to emerging attack vectors, making them an ideal candidate for bolstering web application security.

ML-based solutions have gained attention for improving XSS attack detection. Brilliant results have been achieved by models in detecting XSS attacks, Table 1 shows some of the models and their accuracy implemented by researchers in the field over the past three years.

The values in Table 1 reflect total threat coverage. However, the performance estimation is a big challenge for those models to mitigate realistically the existence of an XSS attack. A set of limitations is considered for justifying the persistent significance of XSS vulnerabilities in the web application security landscape despite efforts to further enhance ML models and detection mechanisms.

The limitations are referred to some reasons:

- ML-based approaches often focus on enhancing classic performance metrics (e.g., accuracy, precision, recall, F1…) properly to reach the best results. Nevertheless, it skipped other important aspects, such as the ability of the model to handle different flavors of XSS (Stored, Reflected, DOM), its adaptability to mitigate new payloads, and its interpretability to validate predictions [5]
- The quality of the preprocessing XSS data can impact significantly the effectiveness and performance of ML-based approaches for XSS attack detection [6]. The main focus of researchers in XSS mitigation is typically on the model regardless of how data was prepared to be processed, especially when considering the large dimensionality of XSS data.

To address the above-given issue, an adversarial evaluation approach is proposed in this article. The motivation for using this evaluation is to provide precious insights into the model's performance, limitations, and effectiveness in real-world scenarios. An evaluating ML-based system for XSS detection works on the concept of subjecting ML models to rigorous assessment, it aims to measure its ability to accurately distinguish between benign user inputs and malicious contents.

One way to get these ML-based models assessed is by altering dataset. Introducing different datasets with different characteristics helps highlight any biases that may occur when the model is exposed to different types of data, and when it is deployed in a diverse environment.

The main contributions in this paper can be presented as below:

- It focuses on performed pre-existing ML-based models for XSS detection, by highlighting their gaps and limitations.

Table 1 Performance metrics of approaches in terms of accuracy

Work	[1]	[2]	[3]	[4]
Accuracy	98,54%	99,75%	99,80%	99,96%

- It creates a competitive framework to assess performance model's by altering dataset.
- It helps future research papers to focus on several aspects before turning their ML-based models to make a balance between accuracy, coverage, and data.

The remainder of this paper is organized as follows. In section II, we give an overview of some research works related to different techniques in evaluating ML-based models for XSS detection. In section III, we present the methodology of our evaluating approach. Our research analysis and findings are presented in section VI. Finally, we conclude our paper with future directions in section V.

2 Related Work

Advanced ML-based models are being developed by researchers in the field of XSS detection and web application security, including single, ensemble, and deep learning methods. Evaluating these models can be conducted by their own authors or by other studies. An overview of methods that assess pre-existing ML-based models for XSS detection is provided in this section.

- The commonly used method to evaluate an ML model is by focusing on its performance's metrics. The majority of ML-based models for XSS detection work to achieve the best results of accuracy and precision etc.… Thus, provides a comparison between multiple models determining the robust one [1, 3, 4, 7–9]
- Author in [10] uses another way to investigate ML-based models by confusing several metrics. Three supervised ML systems are used in this work, and validated with XSS-Attacks-2019 dataset. The system's efficiency was assessed in terms of confusion matrix analysis (i.e., the number of False Positive, True Positive, False Negative, and True Negative), detection accuracy, detection precision, detection sensitivity, and detection time.
- [6] reviews diverse dimensions of pre-existing ML/DL models for XSS detection. The study explores preprocessing steps for XSS detection (e.g., data cleaning, feature selection, dimensions reduction, and balancing data), most ML/DL models used in the field, and evaluation approaches employed to validate those models besides performance metrics including dataset types, data splitting, and cross-validation. The findings extend to consider the impact of data preprocessing techniques in addition to standard performance metrics in evaluating ML models.
- [11] compares two neural network models combined with static approaches for XSS detection in PHP and Node.js code. Each model is evaluated with two different datasets. Datasets are splitting into training, validation, and test sets to select the best models that could be re-assessed one more iteration using the test sets.
- [12–14] uses cross-validation method to evaluate their models. This method aims to divide the dataset into k subsets, k-1 subsets are reserved for the training phase while the remaining subsets are for the testing phase. Researchers in [12] and [13]

evaluate their approaches using k = 10 subsets, repeating the process 10 times to obtain an accurate estimate of the model's performance.

- Another evaluation method for ML-based models is used in the field of Internet of Things (IoT) network intrusion detection [15]. This work employs a cross-dataset validation technique, that uses new datasets to evaluate different ML models and ensure their performance. Cross-dataset validation involves the assessment of the models across multiple datasets. Thus, it provides training the model in one dataset and testing it in another dataset.

3 Methodology

This research presents an assessment system of pre-existing ML-based models that can mitigate the XSS attacks launched against websites/web applications. Our methodology to establish this assessment system is decomposed into six modules (Fig. 2). This includes two phases: the first phase with a model selection module, performance metrics definition module, and model ration module. The second phase includes data selection module, data preprocessing module, and cross-dataset validation module.

3.1 Phase 1: Model Analysis

Existing ML-based models demonstrate great potential for solving XSS. This phase involves understanding the model's architecture and approach, investigating into the technical specifics of its ability to identify and distinguish XSS attacks, highlighting gaps and identifying areas that need more attention.

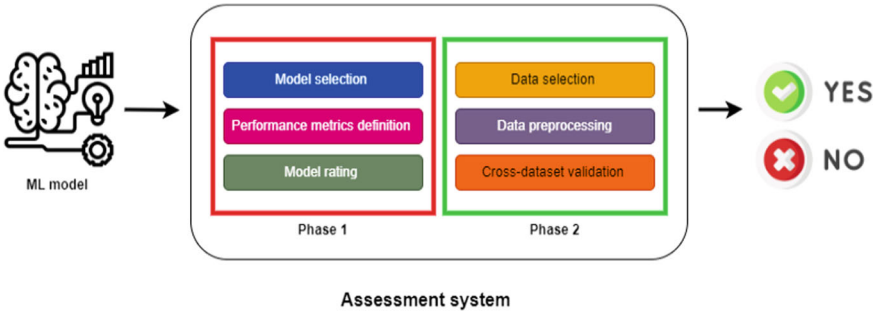


Fig. 2 ML-based models for XSS detection assessment system diagram

3.1.1 Model Selection Module

The first module of our methodology is to select the target ML-based model, discover underlying principles through documentation and research papers. A comprehension of data preprocessing methods, feature extraction, and feature selection choice is required to establish a solid foundation for the subsequent evaluation phase. Various ML models have been proposed to handle XSS in the literature, popular ones include Random Forest (RF), Decision Tree (DT), Naïve Bayes (NB), Support vector machine (SVM), Recurrent Neural Network (RNN), Convolutional Neural Network (CNN), Long-Short Term Memory (LSTM) etc.… others can combine them evidently with metaheuristic algorithms in particular Genetic algorithms.

3.1.2 Performance Metrics Definition Module

This module begins with a comprehensive analysis of the ML-based model's performance metrics. This involves a global view of its capabilities. The use of the confusion matrix involves understanding the model and its performance of classification. Accuracy, precision, recall, and F1-score are calculated to provide insights into the model's ability to detect XSS attacks effectively. Accuracy refers to the measure of how well a model is performing, it represents the proportion of correct predictions made by the model out of total predictions. However, in addition to accuracy, we are dependent on several performances' metrics; precision refers to the measure of actually positive predictions out of all the total positive predicted, it indicates the model's ability to avoid false positives; recall also known as sensitivity refers to the measure of actually positive predictions out of all the total actual positive, it indicates the model's ability to detect true positives; F1-score balances between precision and recall. In this research, we are mainly focusing on XSS mitigation models owning the best performance rates, we describe their strengths and weaknesses within our evaluation system modules.

3.1.3 Model Ration Module

The purpose of this module is to review brilliant pre-existing ML-based models for XSS detection. This leads to highlight limitations that degrade their performance thus striking the right balance between accuracy and efficiency. Five evaluation criteria are defined in light of this objective:

1. EC1. The functionality of the model considering the cost of implementing and maintaining
2. EC2. The specificity of the model regarding different types of XSS attacks (Stored, Reflected, DOM)
3. EC3. The ability of the model to mitigate XSS attacks in both client-side and server-side

Table 2 ML-based models for XSS detection Evaluation Criteria table

	Evaluation Criteria	Score		
		0	1	2
EC1	non-functional = 0/functional = 2			
EC2	unspecified type = 0/detect 2 XSS types = 1/detect all XSS types = 2			
EC3	detect in client or server side = 1/detect in client and server side = 2			
EC4	non-robust against new XSS = 0/robust against new XSS = 2			
EC5	non-interpretable = 0/interpretable = 2			

4. EC4. The robustness of the model against advanced and sophisticated XSS payloads
5. EC5. The interpretability of the model to validate its predictions

The quality of a model in our study depends upon the score obtained according to the evaluation factors (Table 2), even though its accuracy of predictions. A model is marked out of 10. We mentioned that the evaluation criteria used in this paper are based on our survey outcomes about intelligent systems for XSS attack detection.

Each evaluation criteria (EC) admits a value equal to 0, 1, or 2. EC1 refers to the fundamental functionality of the model (i.e., the ability of the model to usefully execute, process inputs, follow the intended logic, and produce the expected output without encountering critical errors). For this criterion, we assume two values, 2 if functional; and 0 otherwise. XSS attacks can be categorized into different flavors depending on how the malicious scripts are injected into web applications and executed in user's browsers, EC2 refers to the ability of the model to mitigate all XSS types. Some models do not specify the type they are dealing with, for this reason, we assume 0 if it does not indicate the type, 1 if it detects two XSS types, and 2 if it detects all types. EC3 refers to the ability of the model to mitigate XSS attacks on the client and/or server side. For this criterion, we assume two values, 1 at least the model detects in client or server side; 2 if it detects in both. The effectiveness of a model against new XSS payloads is assessed through EC4. This criterion accepts two values, 2 if robust against both known and novel XSS payloads; and 0 otherwise. EC5 assesses the interpretability of a model which refers to the transparency of the model to understand and explain its predictions and decisions. Same as EC1 and EC4 we admit two values for this criterion, 2 if interpretable; and 0 otherwise.

3.2 Phase 2: Model Assessment

One of the rigorous ML-based models' evaluations is altering the dataset. It might ensure the model's robustness across different characteristics, variant data distributions, and scenarios. In this paper, we evaluate approved models that are performing well in the current state of the art. A stable model proves resilience against issues,

influences, and open challenges. An unstable model may illustrate false positives, false negatives, and more errors.

3.2.1 Data Selection Module

This module refers to the process of determining the appropriate data used for ML-based models' validation. Several datasets exist for XSS attack detection in the literature, gathered from GitHub, Xssed, OWASP, security forums, etc.… The choice of the dataset considers data diversity (various sources, domains, contexts), data quality (accurate, complete, valid), data volume (adequate to train and test the model) and data representativeness (balance between benign and malicious inputs).

3.2.2 Data Preprocessing Module

A process of preparing, cleaning, and organizing is needed to be applied to the dataset to simplify the complex data form into an understandable form. This module ensures consistent and normalized data presented to the model in the training phase. The process contains data cleaning, feature extraction, and feature selection.

The cleaning data stage involves removing noisy data included by the attacker to avoid detection systems, decoding the mutated or encoded script made by the attacker to hide the malicious content, and segmentation to decompose the script or the payload into smaller units (e.g., phrases, words, characters). The feature extraction aims to obtain the most informative features in raw data. In the field of XSS mitigation, features are extracted from payloads, HTTP requests, HTTP responses, URL parameters, and JavaScript files then represented in the appropriate format that is adapted for the model. The feature selection consists of finding a subset of relevant features in a high-dimensional space of the original dataset. The choice of features depends on the model, dataset, and the objective of the XSS detection system.

3.2.3 Cross-Dataset Validation Module

A robust XSS detection system will guarantee the ability to accurately detect attacks. Assessing its generalization capabilities and performance across various real-world scenarios on another dataset that it has not been trained on, may verify concretely its performance to offer robust protection. Our method resorts to validate ML-based model through dissimilar and unseen datasets. We prepared and processed the new dataset in the same steps as the original dataset, ultimately applying the model to the new dataset in line with monitoring its performance metrics.

4 Results and Discussion

Our assessment system was applied to the model proposed by [3] for presentation in this paper, but it can also be extended to other models. The work presents an approach based on reinforcement learning (RL) combined with genetic algorithms (GA), and threat intelligence to mitigate XSS attacks.

4.1 Phase1: Model Analysis

This initial step presents the motivation to select this module, its main architecture, and its specifics and particularities.

- Model selection module: the choice of the above model was not arbitrary. There are some reasons which motivated us to select this work. (1) merging ML approaches with GA has provided significant benefits to resolve XSS in the current state of the art, (2) the improved accuracy compared with other methods including ensemble approach, SVM, NB, DT, RF, and Logistics regression, and (3) the white-box nature of the model ensures a clear and interpretable insights into the decision-making process.
- Authors in [3] used GA to generate potential solutions to the problem as a starter step, those solutions represent an initial population of chromosomes or individuals. The quality of each chromosome is evaluated by the fitness function. Parent selection is performed to determine which chromosomes will be used for reproduction. New chromosomes are created through crossover and mutation to obtain the fittest and best individuals used in further analysis and reinforcement learning to train the XSS detection model. GA can declare a payload as vulnerable or not by comparing the best chromosomes, in case of failure statistical analysis (SA) is applied subsequently to gather more information about the payloads and distinguish between vulnerable and non-vulnerable ones. RL is employed in this work to train the model for detecting XSS attacks with more flexibility and adaptability.
- Performance metrics definition module: The approach achieves a significant accuracy rate of 99.75% on the original dataset and 99.89% after random injection of malicious payloads.
- Model ration module: Table 3 represents the five evaluation criteria concerning the above-discussed model.

From Table 3, this model is rated 6/10 according to our criteria which is performant, three criteria EC1, EC3, and EC4 are met, but EC2 and EC5 are not. The model addresses existing XSS detection technique limitations through its best accuracy, functionality, coverage, and robustness against new XSS payloads. However, this work did not specify what type of XSS it can detect, and more efforts could be made for interpretability allowing end-users to validate the model's results.

Table 3 Evaluation criteria of [3] model

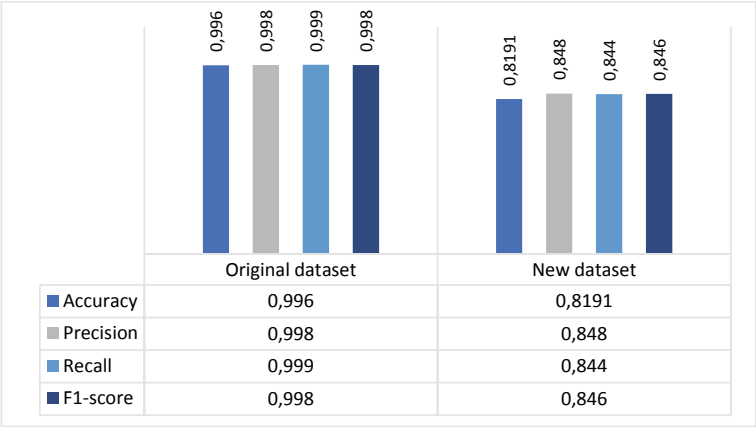
	Evaluation Criteria	Score		
		0	1	2
EC1	non-functional = 0/functional = 2			*
EC2	unspecified type = 0/detect 2 XSS types = 1/detect all XSS types = 2	*		
EC3	detect in client or server side = 1/detect in client and server side = 2			*
EC4	non-robust against new XSS = 0/robust against new XSS = 2			*
EC5	non-interpretable = 0/interpretable = 2	*		

4.2 Phase 2: Model Assessment

This step explores the model's stability concerning variant data distributions and scenarios. We designed the experimental environment in Java 1.8 using IntelliJ IDEA 2023.2.2 over a 64-bit Windows 10 operating system.

- Data selection module: The new dataset used is collected from GitHub to validate the model. it contains normal and vulnerable contents collected from GitHub and provided by other XSS detection approaches. The dataset is divided into vulnerable payloads file and non-vulnerable payloads file to be used in the testing phase.
- Data preprocessing module: Cleaning, removing noisy data, and organizing are applied to our dataset for a reliable foundation. The process of extracting and selecting features is integrated with the model.
- Cross-dataset validation module: In this module, we are following the same steps proposed in the evaluated approach [3] to train the model. 30 features are extracted from payloads and previously used by [1] (e.g., input size, cookie, alert, script, etc.…). Some features are systematically associated with each other despite variations in the payload. After extracting the features, a separate file for each feature is created (i.e., each file contains only payloads that have that specific feature). Features files are converted to binary representation, and all duplicate payloads are removed to improve the model's performance. The non-redundant binary representations of each feature were used as input for the GA to get fit chromosomes. GA generates the fittest then the best chromosomes and patterns for vulnerable and non-vulnerable data. The data was processed to achieve the fittest and best chromosomes, which were then used to mitigate XSS attacks. RL is performed in the second experiment by modifying patterns of different features.

The model was performed using 70% and 30% of the new dataset respectively as a train set and test set. We repeated this process 10 times. The results are described in (Figure 3). As can be deducted from the figure, the performance of the model has declined compared to when it is performed within the dataset. The false positive rate (FPR) is approximately 28.61% which is very important, indicating that the non-vulnerable payloads are falsely detected as vulnerable by the model. These results



	Original dataset	New dataset
Accuracy	0,996	0,8191
Precision	0,998	0,848
Recall	0,999	0,844
F1-score	0,998	0,846

Fig. 3 Performance evaluation analysis for the model with the original and the new dataset

could be linked to an overfitting problem particularly due to the nature of the dataset which is especially designed for DL-based models, or maybe lack of generalization to unseen data might justify these results.

5 Conclusion and Future Directions

Currently, several ML approaches provide shiny results in XSS detection by analyzing web application traffic and identifying malicious behaviors. How to realistically prove, verify, and validate the model's performance and efficiency against XSS; This is what this paper research presents through our assessment system. Two phases are required for ML-based models' evaluation, the first phase concerns the model's architecture evaluation according to five given criteria and the second phase relates to the model's resilience against unseen data distributions. Applying the assessment system to a model proposed by [3] demonstrates some limitations either regarding coverage, integrity, and interpretability or concerning overfitting, generalization, and FPR. In the future, we will adopt more techniques in the evaluation process. By altering multiple datasets for both train and test sets, considering new attack patterns, and comparing the model's behavior within datasets.

References

1. Zhou, Y., Wang, P.: An ensemble learning approach for XSS attack detection with domain knowledge and threat intelligence. Comput. Secur. **82**, 261–269 (2019). https://doi.org/10.1016/j.cose.2018.12.016

2. Mereani, F.A., Howe, J.M.: Detecting cross-site scripting attacks using machine learning. In: Advances in Intelligent Systems and Computing. Springer Verlag pp. 200–210 (2018)
3. Tariq, I., Sindhu, M.A., Abbasi, R.A., et al.: Resolving cross-site scripting attacks through genetic algorithm and reinforcement learning. Expert Syst. Appl. **168**,(2021). https://doi.org/10.1016/j.eswa.2020.114386
4. Alqarni, A.A., Alsharif, N., Khan, N.A., et al.: MNN-XSS: Modular neural network based approach for XSS attack detection. Computers. Mater. Contin. **70** (2022). https://doi.org/10.32604/cmc.2022.020389
5. Et-Tolba, M., Hanin, C., Belmekki, A.: Intelligent systems for XSS attack detection: A brief survey. In: 2023 International Wireless Communications and Mobile Computing, IWCMC 2023 (2023)
6. Thajeel, I.K., Samsudin, K., Hashim, S.J., Hashim, F.: Machine and Deep Learning-based XSS Detection Approaches: A Systematic Literature Review. J. King Saud Univ.-Comput. Inf. Sci. **35**, 101628 (2023). https://doi.org/10.1016/J.JKSUCI.2023.101628
7. Fang, Y., Li, Y., Liu, L., Huang, C.: DeepXSS: Cross site scripting detection based on deep learning. In: ACM International Conference Proceeding Series. Association for Computing Machinery, pp. 47–51 (2018)
8. Yamazaki, K., Kotani, D., Okabe, Y.: Xilara: An XSS filter based on HTML template restoration. In: Lecture Notes of the Institute for Computer Sciences, pp. 332–351. LNICST. Springer Verlag, Social-Informatics and Telecommunications Engineering (2018)
9. Chen, X.L., Li, M., Jiang, Y., Sun, Y.: A comparison of machine learning algorithms for detecting XSS attacks. In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). Springer Verlag, pp. 214–224 (2019)
10. Abu Al-Haija, Q.: Cost-effective detection system of cross-site scripting attacks using hybrid learning approach. Results in Engineering **19**,(2023). https://doi.org/10.1016/j.rineng.2023.101266
11. Maurel, H., Vidal, S., Rezk, T.: Statically Identifying XSS using Deep Learning (2021)
12. Gogoi, B., Ahmed, T., Saikia, H.K.: Detection of XSS Attacks in Web Applications: A Machine Learning Approach. Int. J. Innov. Res. Comput. Sci. & Technol. 9, 1–10 (2021). https://doi.org/10.21276/ijircst.2021.9.1.1
13. FEMTO-ST Institute, Ecole centrale des arts et manufactures (France), École nationale supérieure des mines de Paris, et al Proceedings, 16th IEEE International Conference on High Performance Computing and Communications, HPCC 2014 ; 11th IEEE International Conference on Embedded Software and Systems, ICESS 2014 ; 6th International Symposium on Cyberspace Safety and Security, CSS 2014 : 20–22 August 2014, Paris, France
14. Kascheev, S., Olenchikova, T.: The detecting cross-site scripting (XSS) using machine learning methods. In: Proceedings—2020 Global Smart Industry Conference, GloSIC 2020. Institute of Electrical and Electronics Engineers Inc. pp 265–270 (2020)
15. Farah, A.: Cross Dataset Evaluation for IoT Network Intrusion Detection (2020)

Integrating Artificial Neural Networks and Support Vector Machines Machine Learning Algorithms for Advanced Credit Card Fraud Detection

Oussama Ndama, Ismail Bensassi, and El Mokhtar En-Naimi

Abstract In recent years, rising fraudulent activities have inflicted substantial financial losses across industries, with credit card fraud detection posing persistent challenges due to the diverse techniques employed by fraudsters. This study, an extension of prior research (Ndama, Oussama, and El Mokhtar En-Naimi. "Credit Card Fraud Detection Using SVM, Decision Tree and Random Forest Supervised Machine Learning Algorithms." International Conference on Big Data and Internet of Things. Cham: Springer International Publishing, 2022; Oussama Ndama and El Mokhtar En-Naimi. 2023. Optimisation and Resampling methods for Handling Imbalanced Datasets in Credit Card Fraud Detection using Artificial Neural Networks. In Proceedings of the 6th International Conference on Networking, Intelligent Systems & Security (NISS '23). Association for Computing Machinery, New York, NY, USA, Article 23, 1–10. https://doi.org/10.1145/3607720.3607745), addresses the issue of imbalanced datasets in fraud detection by leveraging the Synthetic Minority Over-sampling Technique (SMOTE). The research explores hybrid methodologies by integrating artificial neural networks with Support Vector Machines (SVM) using various kernels, building upon the initial study. Evaluating 284,807 anonymized transactions, the study emphasizes the comparative performance of the hybrid approach with SMOTE, providing insights into its effectiveness in mitigating challenges associated with imbalanced datasets in credit card fraud detection.

Keywords Credit card fraud detection · Imbalanced datasets · SMOTE (Synthetic Minority over-sampling Technique) · Artificial neural networks (ANN) · Support vector machines (SVM) · Hybrid models

O. Ndama (✉) · I. Bensassi · E. M. En-Naimi
DSAI2S Research Team, C3S Laboratory, FST of Tangier, Abdelmalek Essaâdi University, Tetouan, Morocco
e-mail: oussama.ndama@etu.uae.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_36

1 Introduction

The current business environment has undergone an unparalleled transition towards digital transactions, fundamentally transforming worldwide financial systems. Nevertheless, this advancement in technology has given rise to a widespread problem: the concerning increase in fraudulent behavior, specifically in relation to credit card transactions. It is crucial to promptly and efficiently identify and stop such deceitful activities in order to maintain trust and honesty in global financial systems. In the face of this growing danger, conventional detection technologies have faced difficulties in keeping up with the ever-advanced strategies used by fraudsters. The need for strong and flexible fraud detection techniques has never been more crucial. The deficiencies of current methods emphasize the urgent requirement for sophisticated, groundbreaking solutions capable of countering the complex patterns inherent in fraudulent credit card transactions. The rise of machine learning techniques in recent years has provided a viable solution for tackling this difficulty. Artificial Neural Networks (ANN) and Support Vector Machines (SVM) have emerged as prominent candidates, each offering unique benefits in predictive modeling and classification tasks [1, 2].

This study aims to address the urgent problem of credit card fraud by utilizing the combined capabilities of Artificial Neural Networks (ANN) and Support Vector Machines (SVM). The goal is to create an advanced hybrid technique that combines various methodologies in a way that strengthens the detection process. The proposed hybrid model aims to improve the efficiency and accuracy of credit card fraud detection systems by combining the adaptability and self-learning features of artificial neural networks with the structural robustness and optimal margin separation of support vector machines [3]. The main objective of this study is to present a new framework that combines the capabilities of Artificial Neural Networks (ANN) and Support Vector Machines (SVM) in a mutually beneficial way, taking advantage of their complementing characteristics. This research intends to demonstrate the superior abilities of the hybrid ANN-SVM model in detecting fraudulent activities in credit card transactions through thorough investigation and comparison analysis. This integration not only enhances the accuracy of fraud detection but also adds to the continuous development of machine learning approaches in the fight against financial fraud.

The next sections will outline the methodology, experimental setting, and a thorough evaluation of the hybrid model's effectiveness in identifying fraudulent patterns in credit card transactions. This will enhance the range of advanced fraud detection methods available.

2 Related Works

Our study thoroughly examines the field of credit card fraud detection using Machine Learning and Deep Learning techniques. We carefully analyze and compare a wide range of methodologies, including hybrid models that combine SVM and ANN, the novel application of SMOTE, ensemble models, bankruptcy prediction models, and various machine learning algorithms. This examination compares and examines the effectiveness and suitability of certain approaches in different situations, providing detailed and subtle insights. Hybrid models exhibit a wide range of performance capabilities, however the incorporation and efficacy of Support Vector Machines (SVM), Artificial Neural Networks (ANN), and Synthetic Minority Over-sampling Technique (SMOTE) varies in various research, suggesting a distinct influence on accuracy, precision, and recall measurements. The comparison between Zahra Faraji's ensemble model [4] and Tuong Le et al.'s HAOC approach [5] reveals distinct strengths: Faraji's model, which focuses on logistic regression, decision trees, and XGBoost, performs exceptionally well in terms of precision and recall. On the other hand, HAOC, utilizing a cost-sensitive learning framework along with oversampling, provides a specific solution for addressing class imbalance. The importance of Sivanantham et al.'s vote classifier methodology [6], namely in finding key variables for fraud detection, becomes evident when compared to other methodologies. In addition, our analysis encompasses deep learning techniques, specifically comparing Alharbi et al.'s innovative text-to-image conversion method [7] with Hashemi, Mirtaheri, and Greco's Bayesian optimization for class weight-tuning [8]. This comparison sheds light on various strategies used to tackle online credit card fraud. The review concludes by examining the DEEP-BO algorithm developed by Cho et al. [9] for optimizing hyperparameters in deep neural networks, emphasizing its superior performance compared to alternative deep learning approaches. This extensive comparison analysis not only identifies the advantages and disadvantages inherent in each methodology but also sets the stage for future research to improve these methodologies, increasing their reliability and effectiveness in detecting credit card fraud.

3 Our Proposed Framework

3.1 Architecture

The foundation of our hybrid approach for credit card fraud detection leverages the capabilities of two robust machine learning algorithms: the Artificial Neural Network (ANN) and the Support Vector Machine (SVM). Our approach begins with meticulous data preprocessing, involving normalization, feature engineering, and data

balancing, ensuring data integrity and optimal information representation. The architecture capitalizes on the strength of ANN [10], utilizing it for initial feature extraction. Extracted features undergo optional hyperparameter tuning before training the SVM classifier, serving as the primary model for fraud classification. This integrated architecture seamlessly combines the strengths of ANN and SVM [11], offering an innovative solution to credit card fraud detection challenges. The ANN's pattern extraction proficiency complements the SVM's precise classification, resulting in a robust hybrid model for accurate fraud prediction and heightened security in financial transactions. The architectural design incorporates SVM's flexibility with various kernels and fine-tuning hyperparameters to optimize the model's predictive power [12]. This adaptive mechanism allows the SVM model to flexibly adapt to different data types, enhancing its versatility in capturing complex patterns. Leveraging multiple kernels—such as linear, radial basis function (RBF), and polynomial—and iterative adjustment of hyperparameters, the architecture strengthens the predictive capabilities of the hybrid model. This adaptation and selection process optimize the model's performance in accurately classifying fraudulent transactions while minimizing false positives and false negatives, refining the efficacy and reliability of the entire fraud detection system.

3.2 Algorithms

3.2.1 Artificial Neural Network (ANN)

The Artificial Neural Network (ANN) serves as the cornerstone in our fraud detection framework, harnessing its exceptional capabilities in learning complex patterns within the dataset. This powerful model, inspired by the human brain's neural network, comprises multiple interconnected layers of nodes (neurons) that process information. These interconnected layers enable the ANN to uncover intricate relationships within the dataset, extracting meaningful features vital for fraud detection. The network's architecture involves input, hidden, and output layers, with each layer playing a crucial role in pattern extraction and classification. The backpropagation algorithm fine-tunes the connections between these layers by adjusting the weights, allowing the ANN to learn and improve its predictions over time [13].

3.2.2 Support Vector Machine (SVM)

In contrast to the ANN, the Support Vector Machine (SVM) is a robust discriminative model widely recognized for its effectiveness in binary classification tasks. SVM achieves this by identifying the optimal hyperplane that best separates data points into distinct classes [14]. Its strength lies in the utilization of different kernels, such as linear, radial basis function (RBF), and polynomial kernels. These kernels establish diverse decision boundaries in feature space, enabling SVM to adeptly handle

complex, nonlinear relationships within the data. The linear kernel suits linearly separable datasets, the RBF kernel excels in capturing patterns in nonlinear datasets, and the polynomial kernel accommodates complexities through an adjustable degree parameter [15]. Each kernel operates uniquely, contributing to the SVM's overall robustness and versatility in fraud detection by adapting to diverse dataset structures.

3.3 Data Preprocessing

The hybrid credit card fraud detection framework starts with thorough data preprocessing, ensuring the dataset's readiness for subsequent model training and evaluation. This involves essential steps in data cleansing and preparation. The preprocessing process transforms raw data into a more analytically usable form. Key actions include normalizing the 'Amount' column using **StandardScaler** to fit values within [-1, +1], creating a new column 'NormalizedAmount' for standardized data representation. Less relevant columns like 'Amount' and 'Time' are care-fully removed to enhance the model's focus on salient features. Data integrity is maintained by examining and removing potential duplicate records. The dataset is then divided into feature set ('X') and target variable ('y'), representing fraudulent and non-fraudulent transactions. These steps lay the foundation for the sub-sequent deployment of the hybrid framework, ensuring accurate and robust credit card fraud detection.

3.4 Data Resampling

In the next phase of the hybrid framework, we address data imbalance in credit card fraud detection. Here, we introduce the Synthetic Minority Over-sampling Technique (SMOTE). SMOTE acts as an alchemist, infusing the dataset with syn-thetic samples for the minority class. This transformative process orchestrates a dance, rebalancing the dataset in a harmonious rhythm, aligning class distributions for both 'X' and 'y'. This strategic choreography serves as a countermeasure, artfully mitigating the inherent imbalance between fraudulent and non-fraudulent transactions.

At its core, SMOTE has a unique idea behind it. It cleverly fills in the feature space by making new instances that follow the lines connecting existing minority class samples. This smart method boosts the dataset, giving more attention to the less-represented class and helping us understand complex patterns better [16]. After using SMOTE, the resampled feature set 'X_resample' and the updated target variable 'y_resample' are carefully organized into detailed Pandas Dataframes. In this carefully prepared setup, 'y_resample' takes on a crucial role, making sure things are accurate through the 'Class' column. It creates a clear difference between fraudulent and non-fraudulent cases, like a well-organized musical piece. At the same time, the 'X_resample' Dataframe becomes the main focus, with resampled features working smoothly alongside the original ones. This thoughtful arrangement improves the

dataset, getting it ready for the important steps of training and testing the model in the hybrid framework for credit card fraud detection.

3.5 Data Segmentation

In the subsequent phase, we meticulously divide the resampled dataset into training and testing subsets to pave the way for robust model evaluation and validation. This partitioning is executed with precision using the 'train_test_split' method from the scikit-learn library, ensuring a stratified split that maintains the proportional representation of both fraudulent and non-fraudulent instances within the subsets.

The split follows a well-considered 70:30 ratio, where 70% of the data is allocated for training ('X_train' and 'y_train'), while the remaining 30% is earmarked for testing ('X_test' and 'y_test'). This careful balance ensures that the model is exposed to a diverse set of instances during training and testing, contributing to its ability to generalize well to unseen data.

Subsequently, the arrays resulting from this meticulous data partitioning process are transformed into NumPy arrays. This conversion is a preparatory step, setting the stage for the upcoming model training and evaluation within the credit card fraud detection framework. The attention to detail in this partitioning process enhances the reliability and effectiveness of the subsequent model assessments.

3.6 Data Evaluation

The section provides an overview of the hybrid model development and evaluation process for credit card fraud detection. Initially, the Artificial Neural Network (ANN) model is trained on the pre-processed data, consisting of 16 units in the in-put layer, followed by two hidden layers with 24 and 1 units, respectively. This model is compiled with the Adam optimizer and binary cross-entropy loss function to predict the activation of fraudulent transactions, ensuring robustness in identifying anomalous activities. Once trained, the ANN model is utilized to derive feature vectors from the training dataset. These vectors form the foundation for training the Support Vector Machine (SVM) model in the subsequent phase. The SVM is strategically fine-tuned using hyperparameters, such as the regularization parameter 'C', the kernel coefficients 'gamma,' the kernel type, and the polynomial degree, via 'GridSearchCV' to obtain the best-performing configuration.

Following the SVM's successful training, the hybrid model's efficacy is evaluated using the test dataset. Utilizing the ANN model, feature vectors are extracted from the test set and employed in predicting fraudulent transactions using the optimized SVM model. The evaluation includes various performance metrics, such as accuracy, precision, recall, and the F1 score. Furthermore, confusion matrices are generated

for both the test and full datasets, visualizing the model's performance in classifying fraudulent and non-fraudulent transactions.

Finally, the obtained results are tabulated into a comprehensive Dataframe for comparative analysis with other models, providing insights into the hybrid model's performance in credit card fraud detection.

4 Results and Discussion

The results of the model evaluation indicate the effectiveness of different hybrid configurations of ANN-SVM in credit card fraud detection. Each configuration provides valuable insights into the trade-offs among different hyperparameters, resulting in diverse levels of accuracy, precision, recall, and F1 scores.

In our meticulous evaluation of credit card fraud detection, particularly by highlighting the superior efficacy of SVM kernel-based models. The novel insights from our study, especially regarding the ANN-SVM Linear with C = 0.1, ANN-SVM RBF with Gamma = 10.0, and ANN-SVM Poly with degree 4 models, represent a substantial advancement over existing methodologies. Our results show that these models perform better than any others in both test and full datasets, beating the standards set by earlier research [4–9]. In term of recall, precision, false negative rates, and overall accuracy, our models set new standards. For instance, the ANN-SVM Linear model exhibited an exceptional recall of 99.96% and a remarkably low false negative rate of 0.038%, outperforming the models studied [4] and others. Similarly, the ANN-SVM RBF model, with its zero false negatives and balanced recall of 99.99%, signifies a breakthrough in achieving high precision and accuracy, a feat not paralleled in earlier studies. The ANNSVM Poly 4 model further solidifies the superiority of our approach with its impressive recall and precision rates. Extending the evaluation to the full dataset, these models continued to demonstrate excellence in recall, precision, false negative rates, and accuracy. The ANN-SVM RBF model with Gamma = 10.0 maintained a perfect recall of 100% and zero false negatives, underscoring its robust performance, precision 99.99%, and impressive accuracy 99.92%. Both the ANN-SVM Linear and ANN-SVM Poly 4 models sustained high recall (99.97% and 99.95%, respectively), precision metrics, and accuracy (99.86% and 99.89%, respectively). Notably, the ANN-SVM RBF with Gamma = 10.0 emerged as the most promising model, excelling in all metrics, making it an optimal choice for real-world credit card fraud detection applications

Contrasting these results with existing research, we notice that while studies like [4] and [5] have laid a solid foundation using XGBoost, logistic regression, and hybrid approaches like HAOC, our models demonstrate a significant leap forward in terms of accuracy and reliability. Our findings suggest a more nuanced understanding of kernel selection's impact in SVM models, an area that has received less emphasis in previous works. Similarly, our research complements the deep learning advancements proposed by [7] and the hyperparameter optimization techniques by [9], further enriching the field's knowledge base.

Table 1 Result of key performance indicators for the three best models from different kernels among test dataset

Model	Accuracy	FalseNegRate	Recall	Precision	F1 Score
ANN-SVM Linear C = 0.1	0.998916	0.000387	0.999613	0.998223	0.998917
ANN-SVM RBF Gamma = 10.0	0.999194	0.000109	0.999891	0.998501	0.999195
ANN-SVM Poly 4	0.998904	0.000460	0.999540	0.998271	0.998905

This comprehensive comparative analysis and the results of our study underscore the added value of our research in the scientific field of credit card fraud detection. Our models not only address the limitations identified in previous studies but also introduce innovative approaches that significantly enhance detection accuracy and reliability. As such, this research opens new avenues for future studies to build upon, particularly in refining and optimizing SVM kernels and deep learning techniques for more robust and efficient fraud detection systems.

5 Conclusion and Perspectives

Following an extensive exploration and practical application of diverse hybrid models aimed at credit card fraud detection, this research journey has unveiled a significant revelation concerning the effectiveness of amalgamating Artificial Neural Networks (ANN) and Support Vector Machines (SVM). The investigation into various kernels within this hybrid framework has provided insightful revelations regarding their impact on precision and recall. The results underscored the diverse trade-offs observed across different kernels in the ANN-SVM framework. Particularly noteworthy were the RBF kernels with elevated gamma values, which emerged as robust performers in capturing fraudulent activities, albeit with a slight compromise on precision. In contrast, linear and polynomial kernels demonstrated a balance between precision and recall, offering essential flexibility for tailored fraud detection strategies. The deployment of these hybrid models, coupled with a comprehensive evaluation using diverse performance metrics, has illuminated the intricate landscape of credit card fraud detection. The study's outcomes emphasize the versatility and efficiency of the ANN-SVM hybrid framework in addressing the challenges associated with imbalanced datasets in fraud detection. These insights usher in new horizons for customized fraud detection models, promising potential in fortifying the security of financial transactions and establishing a robust defense against fraudulent activities. The hybridization of ANNs and SVMs, especially considering varying kernel configurations, emerges as an advanced and adaptable solution that enhances the capability of fraud identification in real-world scenarios (Tables 1 and 2).

Table 2 Result of key performance indicators for the three best models from different kernels among full dataset

Model	Accuracy	FalseNegRate	Recall	Precision	F1 Score
ANN-SVM Linear C = 0.1	0.998596	0.002114	0.997886	0.550117	0.709241
ANN-SVM RBF Gamma = 10.0	0.998883	0.000000	1.000000	0.605634	0.754386
ANN-SVM Poly 4	0.998647	0.002114	0.997886	0.559242	0.716781

References

1. Tsai, C.-F., Chen, M.-L.: Credit rating by hybrid machine learning techniques. Appl. Soft Comput. **10**(2), 374–380 (2010)
2. Qiang, W.: A hybrid sampling SVM approach to imbalanced data classification. Abstr. Appl. Anal. (2014). Hindawi, 2014
3. Abhimanyu, R., et al.: Deep learning detecting fraud in credit card transactions. In: 2018 systems and information engineering design symposium (SIEDS). IEEE, 2018
4. Faraji, Z.: A review of machine learning applications for credit card fraud detection with a case study. SEISENSE J. Manag. **5**(1), 49–59 (2022)
5. Tuong, L., et al.: A hybrid approach using oversampling technique and cost-sensitive learning for bankruptcy prediction. Complexity (2019)
6. Sivanantham, S., et al.: Hybrid approach using machine learning techniques in credit card fraud detection. In: Advances in Smart System Technologies: Select Proceedings of ICFSST 2019. Springer Singapore, 2021
7. Abdullah, A., et al.: A novel text2IMG mechanism of credit card fraud detection: A deep learning approach. Electronics 11(5), 756 (2022)
8. Seyedeh Khadijeh, H., Leili Mirtaheri, S., Sergio Greco: Fraud Detection in Banking Data by Machine Learning Techniques. IEEE Access 11, 3034–3043 (2022)
9. Hyunghun, C., et al.: Basic enhancement strategies when using Bayesian optimization for hyperparameter tuning of deep neural networks. IEEE Access 8, 52588–52608 (2020)
10. Oludare Isaac, A., et al.: Comprehensive review of artificial neural network applications to pattern recognition. IEEE Access 7, 158820–158846 (2019)
11. Hosseini, S., Zade, B.M.H.: New hybrid method for at-tack detection using combination of evolutionary algorithms, SVM, and ANN. Comput. Netw. **173**, 107168 (2020)
12. Batuwita, R., Palade, V.: FSVM-CIL: fuzzy support vector machines for class imbalance learning. IEEE Trans. Fuzzy Syst. **18**(3), 558–571 (2010)
13. Asha, R.B., Suresh Kumar, K.R.: Credit card fraud detection using artificial neural network. Glob. Transit.S Proc. **2**(1), 35–41 (2021)
14. Amarappa, S., Sathyanarayana, S.V.: Data classification using Support vector Machine (SVM), a simplified approach. Int. J. Electron. Comput. Sci. Eng **3**, 435–445 (2014)
15. Arti, P., Singh Chouhan, D.: SVM kernel functions for classification. In: 2013 International-conference on advances in technology and engineering (ICATE). IEEE (2013)
16. Gyoten, D., Ohkubo, M., Nagata, Y.: Imbalanced data classification procedure based on SMOTE. Total. Qual. Sci. **5**(2), 64–71 (2020)

Enhancing Security in Edge Computing with RSA and Paillier Encryption Scheme

Hamid El Bouabidi, Mohamed EL Ghmary, Salah Eddine Hebabaze, and Mohamed Amnai

Abstract Edge Computing (EC) is a form of computing that utilizes the unique capabilities of devices and networks that are beyond traditional desktops, laptops, or mobile devices. As a result, edge devices are more powerful and more versatile than traditional devices, which enables them to handle more complex tasks and applications. The EC can enhance the performance and reliability of digital products and services while lowering costs. The EC has many advantages, but it also has a number of risks associated with it. This paper reviews previous studies in order to identify concerns regarding security with EC, also, we present the results of our approach for securing edge computing using two partial homomorphic encryption schemes, namely Paillier and RSA. Our research specifically evaluates the performance of our approach by measuring the encryption time, operation sum (Paillier), or multiplication (RSA) on encrypted values, and decryption time.

Keywords Edge computing · Edge computing security · Internet of Things (IoT) · Cloud computing · Fog computing · Mobile edge computing · Homomorphic encryption · RSA · Paillier

H. El Bouabidi (✉) · S. Eddine Hebabaze · M. Amnai
Departement of computer science Ibn Tofaıll, University Faculty of Sciences, Kenitra, Morocco
e-mail: hamid.elbouabidi@uit.ac.ma

S. Eddine Hebabaze
e-mail: salaheddine.hebabaze@uit.ac.ma

M. Amnai
e-mail: mohamed.amnai@uit.ac.ma

M. EL Ghmary
Faculty of Sciences Dhar El Mahraz, Departement of Computer Science, idi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: mohamed.elghmary@usmba.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_37

1 Introduction

The concept of edge computing has emerged as a result of cloud computing issues including excessive latency, inadequate bandwidth, high energy consumption, and data security. Edge Computing (EC) is a novel technology that processes data utilizing local devices such as phones and personal PCs. Healthcare [1] , banking [2], and manufacturing [3] are just a few of the areas that may leverage this kind of computing. EC, by its own nature, increases resource speed and efficiency via the employment of specialized software and hardware [4]. This kind of computing is best suited for operations carried out at the network's edges, where it may provide consumers a quicker or more effective user experience and enhance system performance [5]. Edge computing is not a new concept. It is utilized in critical response time scenarios, playing a significant role in tasks like industrial process control, military applications, and telecommunications. EC is very significant in the Internet of Things (IoT) [6]. A huge number of devices, frequently in faraway places, create data in IoT applications [7]. Data from these devices must be collected, processed, and transmitted to a data center for further examination. Edge computing may speed up processing, use less energy, and reduce the amount of data that has to be sent over the network. [8],[9]. The attack surface has grown significantly as edge computing becomes more widespread, with a rising number of devices storing data. Denial-of-service and data attacks are becoming increasingly widespread. And, as the usage of personal devices and millions of people compelled to work remotely has shown, a diverse workforce makes adequately monitoring and protecting devices much more challenging. Add to it the fact that these devices are linked to other devices that the security team is unaware of, and the danger grows. Edge computing is rapidly evolving, offering enhanced data and application security [10]. As its popularity grows, addressing security risks becomes crucial, with unauthorized access and data breaches being primary concerns [11]. The decentralized architecture, while reducing latency, introduces security challenges, especially in resource-constrained environments [12]. Authentication, access control, encryption techniques, secure communication protocols, and network segmentation are proposed as measures to counter these challenges. Homomorphic encryption emerges as a promising solution [13], allowing secure data processing without decryption, effectively balancing data privacy and processing efficiency. However, its implementation in edge computing requires careful consideration of performance trade-offs and compatibility with existing infrastructure. Ongoing research and development are essential to optimize homomorphic encryption's efficiency for edge computing, ensuring the security, confidentiality, and privacy of data in this evolving technological landscape. This article presents a solution to secure edge computing by ensuring secure communication between the edge device and the edge server

2 Related Works

In the realm of enhancing security in edge computing, several related works have explored the application of different encryption schemes to safeguard sensitive data and address the unique challenges posed by this distributed computing paradigm. Baharami et al. [14] found a way to store data on multiple clouds using a unique method of encryption. They used pseudo-random permutation, or PRPM, which they derived from chaotic systems. These systems mask data by scrambling it through substitution operations performed on mobile devices. This approach makes the process incredibly lightweight and helps maintain privacy. At the Edge nodes access a large volume of data contained within clients' equipment and third party applications. Data gathered is analyzed and processed by edge computing nodes that many people trust. However, the nodes can still be vulnerable to a successful hacking attack. In [15] Authors suggested using homomorphic encryption to solve the problems of data leakage. In [16] authors propose a data privacy scheme based on Hash-Solomon code intelligence protects fragmentary information, but authentication is not considered. To secure communication between two entities, authors in [17] proposed elliptic curve integrated encryption scheme (ECIES), which employs an asymmetric approach to generate a shared key. The mentioned scheme did not adequately describe the specific methodology employed for securing the plain text through Elliptic Curve Cryptography scheme or the process of encoding it into numerical values for utilization in ECC's mapping phase. In [18] authors suggested a storage privacy scheme that uses computational intelligence to preserve data privacy and aggregation, but anonymity is not considered.

3 Securing the Edge Computing Using Two Partial Homomorphic Encryption

Homomorphic encryption is a cryptographic technique that permits processing on encrypted data without requiring it to be decrypted. Partial homomorphic encryption allows only one operation (addition or multiplication) to be performed on encrypted data. Paillier and RSA are two common partial homomorphic encryption techniques.

3.1 Paillier Additive Homomorphic Encryption

Pascal Paillier developed the Paillier encryption technique in 1999 [19]. This technique encrypts with a public key and decrypts with a private key. Since the technique is probabilistic, the encryption of the same plaintext might result in distinct ciphertexts. The Paillier algorithm is homomorphic for addition, which means it can add ciphertexts without decrypting them. The Paillier encryption scheme is a probabilistic asymmetric encryption scheme that supports homomorphic addition operations that can be divided into 3 main steps:

Key Generation:
In the key generation process, two large prime numbers, p and q, are carefully selected. The modulus n is then computed as the product of p and q, while λ is determined as the least common multiple of $(p-1)$ and $(q-1)$. A random value g is chosen from the set of invertible elements modulo n^2, satisfying $g^n \equiv 1 \pmod{n^2}$. The value μ is calculated using the L function, where $L(x) = \frac{x-1}{n}$. μ is the modular inverse of $L(g^\lambda \bmod n^2)$ modulo n. The public key is represented by (n, g), while the private key is λ.
Encryption: for encryption, given a plaintext message m within the set of integers modulo n, a random value r is selected from the set of integers modulo n, and the ciphertext c is computed as $c = g^m \cdot r^n \bmod n^2$.
Homomorphic addition: allows the computation of the sum of plaintexts from two ciphertexts, c_1 and c_2, representing plaintexts m_1 and m_2, respectively. The sum is obtained by multiplying the ciphertexts: $c = c_1 \cdot c_2 \bmod n^2$.
Decryption: finally, in the decryption process, given a ciphertext c, the plaintext message m is obtained by computing $m = L(c^\lambda \bmod n^2) \cdot \mu \bmod n$. These steps collectively illustrate the functioning of a cryptographic scheme based on homomorphic encryption, providing secure operations on encrypted data.

3.2 RSA Multiplicative Holomorphic Encryption

Ron Rivest, Adi Shamir, and Leonard Adleman invented the RSA encryption algorithm in 1977 [20]. It is based on how difficult it is to factor huge numbers. RSA encryption employs both a public and a private key for encryption and decryption. The RSA algorithm is deterministic, which means that it always produces the same ciphertext when encrypted with the same plaintext. The RSA algorithm is homomorphic for multiplication, which means it can multiply ciphertexts without decrypting them. Because of this quality, RSA encryption is useful for privacy-preserving product computations, such as computing the encrypted product of two values. The RSA encryption algorithm comprises three main components that include key generation, encryption, and decryption processes: Ron Rivest, Adi Shamir, and Leonard Adleman introduced the RSA encryption algorithm in 1977 [20]. This algorithm is founded on the complexity of factoring large numbers. RSA utilizes both a public and a private key for encryption and decryption, demonstrating determinism by consistently producing the same ciphertext when encrypting identical plaintext. Notably, RSA is homomorphic for multiplication, allowing the multiplication of ciphertexts without decryption. This property makes RSA valuable for privacy-preserving product computations. The RSA encryption process involves key generation, encryption, and decryption. During key generation, two distinct prime numbers, p and q, are selected, and the public key is represented as (n, e), while the private key is (n, d). Encryption involves computing the ciphertext c from a plaintext message m within the range $0 \leq m < n$ using $c = m^e \bmod n$. Homomorphic multiplication enables the computation of the product of two plaintexts from their ciphertexts, represented as $c = (c_1 \cdot c_2) \bmod n$. Finally, decryption, given a ciphertext c, retrieves the plaintext message m using $m = c^d \bmod n$. In summary, RSA encryption, with its key

generation, encryption, and decryption processes, provides a secure and versatile cryptographic approach. Paillier and RSA are two common partial homomorphic encryption techniques. They are better suited for distinct sorts of calculations, with Paillier encryption excelling at addition and RSA encryption excelling at multiplication. These algorithms are useful for privacy-preserving computing, such as secure two-party calculation.

3.3 Combining Paillier and RSA Encryption for Computations

Our solution aims to enhance security in edge computing by establishing secure communication between edge devices and servers. We've developed an interface to be installed on client devices. This interface encrypts each database table using two partly homomorphic algorithms: the Paillier algorithm for addition and RSA for multiplication. The entire database is encrypted before transmission to the edge and cloud servers. Users can choose columns and operations. The interface identifies the type of calculation (addition or multiplication) and selects the appropriate encrypted table (encrypted with Paillier or RSA) at the edge server. If addition is requested, Paillier homomorphic encryption is performed. For multiplication, RSA is utilized. The encrypted result is sent to the client, which decrypts it using the secret key and displays the clear result. This process is facilitated by the following algorithm.

Algorithm 1 Encrypt Database

Require: A database object DB
Ensure: Two encrypted tables TP and TR
$encDB \leftarrow \emptyset$ // initialize encrypted database
$TR \leftarrow \emptyset$ // initialize RSA table
$TP \leftarrow \emptyset$ // initialize Paillier table
// Retrieve 'T' table from the existing clear database
for all T in DB **do**
  // for each table in the database
  for all cl in T **do**
    // for each column in clear table T
    $cle.name \leftarrow$ encrypt$(cl.name, RSA)$
    TR.insert$(cle.name)$
    TP.insert$(cle.name)$
    for $i = 1$ to cl.length **do**
      TR.cle.val$(i) \leftarrow$ encrypt$(T$.cl.val$(i), RSA)$
      TP.cle.val$(i) \leftarrow$ encrypt$(T$.cl.val$(i), Paillier)$
    end for
  end for
  $encDB$.insert(TR)
  $encDB$.insert(TP)
end for
return $encDB$

Algorithm 1 takes a clear database object *DB* as input and returns an encrypted database *encDB*. For simplification reasons, we assume that the database has only one table *T*. The goal is to encrypt the data in the database using both RSA and Paillier encryption schemes while preserving the data structure. The algorithm starts by initializing three variables: *encDB*, *TR*, and *TP*. *TR* and *TP* will store RSA-encrypted and Paillier-encrypted data, respectively while *encDB* will store the encrypted database. The *T* table is then retrieved from the existing clear database. For each table in the database, the algorithm loops over each column *cl* in the *T* table. The column name is encrypted using RSA and inserted into the *TR* and *TP* tables. For each value in the column, the value is encrypted using RSA and inserted into the *TR* table, and also encrypted using Paillier and inserted into the *TP* table. Finally, the encrypted tables *TR* and *TP* are inserted into the encrypted database *encDB*. The encrypted database encDB is then returned. Overall, this algorithm uses both RSA and Paillier encryption schemes to protect sensitive data in a database while preserving its structure. By encrypting both the column names and values, the algorithm provides an additional layer of security. The algorithm ensures that the encrypted database is structured in the same way as the original database, making it possible to query and analyze the data in the future.

The user first selects a specific database he intends to work on, as illustrated in Fig. 1 Next, the client interface employs a public key to encrypt the selected database. The user then specifies the desired column and operation to be performed. The client interface receives the encrypted outcome from the edge server and utilizes the secret private key to decrypt and display the result for the user When the user requests an addition or multiplication operation, he chooses the column to operate on and the client sends the column ID and operation code (sum/multiplication) to the edge server as shown in Fig. 2. The edge server analyzes the operation code and chooses the appropriate encrypted table, utilizing the appropriate homomorphic

Fig. 1 Client side

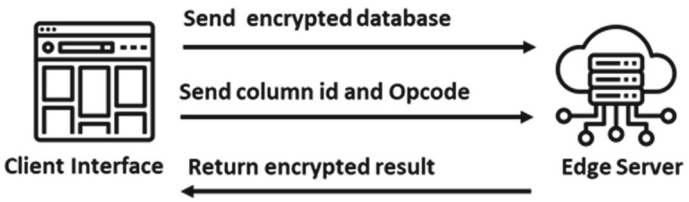


Fig. 2 Server side

Table 1 Encryption time

Key size	Paillier encryption	RSA encryption	Total time
128	2.13	0.91	3.04
256	17.35	5.52	22.87
512	85.68	24.69	110.37
1024	543	135.79	678.79
1536	1686.3	373.212	2059.512
2048	3783.14	890.7	4673.84

Table 2 Homomorphic operation time

Key size	Sum (Paillier)	Multiplication (RSA)
128	0.82	0.79
256	0.75	1.24
512	2.92	1.22
1024	2.13	2.36
1536	2.86	2.58
2048	4.61	5.77

scheme to operate on encrypted values. The RSA homomorphic scheme is used for multiplication operations, while the Paillier homomorphic scheme is used for addition operations. Finally, the edge server sends the encrypted result to the client interface. By using encryption and homomorphic schemes, this system ensures that sensitive data is kept private and secure, while still allowing for necessary computations to be performed on the data. In this study, we present the results of our approach for securing edge computing using two partial homomorphic encryption schemes, namely Paillier and RSA. Our research specifically evaluates the performance of our approach by measuring the encryption time, operation sum (Paillier), or multiplication (RSA) on encrypted values, and decryption time. The findings indicate the feasibility of our approach and demonstrate its potential to provide secure and privacy-preserving computation in edge computing environments (Table 1).

The first column lists the key sizes being used for encryption, and the second and third columns show the time it takes to encrypt with the Paillier and RSA algorithms in milliseconds, respectively. The final column shows the total encryption time, which is the sum of the Paillier and RSA encryption times. It can be seen that as the key size increases, the encryption time also increases for both algorithms. However, the Paillier encryption time grows much faster than the RSA encryption time. For example, when the key size is increased from 128 to 256, the Paillier encryption time increases by a factor of more than 8, while the RSA encryption time increases by only about 6 times. At the largest key size of 2048, the Paillier encryption time is about 4 times slower than the RSA encryption time (Table 2).

Table 3 Result decryption time

Key size	Paillier decryption	RSA decryption	Total time
128	1.02	0.99	2.01
256	1.52	1	2.52
512	2.99	1.35	4.34
1024	12.81	11.18	23.99
1536	17.14	27.14	44.28
2048	40.18	65.66	105.84

The table shows the average time taken to perform homomorphic addition and multiplication operations using the Paillier and RSA schemes, respectively, for different key sizes. The key size indicates the strength of the encryption and decryption processes, and a larger key size generally provides higher security at the cost of increased computational time. Based on the table, we can observe that the time taken for homomorphic operations increases with the increase in key size. For example, for a key size of 128, the time taken for both addition and multiplication is less than 1 millisecond. However, for a key size of 2048, the time taken for multiplication is more than 5 milliseconds. We can also observe that the time taken for multiplication operations using the RSA scheme is generally higher than that for addition operations using the Paillier scheme. This is because the RSA scheme involves more complex mathematical operations compared to the Paillier scheme. Overall, the table provides useful information for optimizing the performance of the homomorphic encryption system. Based on the table, the system can be configured to use the appropriate homomorphic scheme based on the operation requested and the key size selected to balance the trade-off between security and computational time (Table 3).

The presented table shows the decryption time of our approach for different key sizes. The Paillier and RSA decryption times, as well as the total decryption time, are measured in milliseconds. As we can see from the table, the decryption time increases with the key size. This is expected since larger keys require more computational resources to decrypt. The Paillier decryption time is generally faster than the RSA decryption time, except for the 512-bit key size. This is because the RSA decryption algorithm is faster for smaller key sizes, while the Paillier decryption algorithm becomes faster for larger key sizes. Overall, the total decryption time is the sum of the Paillier and RSA decryption times. Therefore, it is important to select an appropriate key size that balances security with performance. By using encryption and homomorphic schemes, our approach ensures that sensitive data is kept private and secure, while still allowing for necessary computations to be performed on the data. In today's world of ever-increasing amounts of data, it is essential to ensure the privacy and security of sensitive information. One way to achieve this is through the use of encryption and homomorphic encryption schemes. In the system described, the client interface encrypts the database using the public key and sends the encrypted database to the edge server. When a user requests an addition or multiplication

operation, they choose the column to operate on, and the client sends the column ID and operation code (sum/multiplication) to the edge server. The edge server then analyzes the operation code and selects the appropriate encrypted table. To perform multiplication operations, the edge server utilizes the RSA homomorphic scheme, while for addition operations, the Paillier homomorphic scheme is employed. After performing the requested operation, the edge server sends the encrypted result back to the client interface. The client interface decrypts the result using the private key and displays it to the user, ensuring that sensitive data remains secure throughout the entire process. This system provides a secure and efficient method for data processing while maintaining privacy and security.

4 Conclusion

Edge computing is a new approach to computing that focuses on handling applications at the outer edges of the network. By providing resources such as storage, network, and processing power, edge computing can help support latency-sensitive applications while reducing pressure on the core network. It's imperative to explore and study the issues of edge nodes related to high performance, secure collaboration, and intelligent management. This is because edge nodes have limited computing and storage space; they're less secure than cloud computing centers. This article presents an overview of Edge Computing (EC), the risks it faces, and a proposed approach that uses homomorphic encryption to securely perform operations on the data. The client encrypts the database using the public key and sends it to the edge server. The server then performs the requested operation (addition or multiplication) on the appropriate encrypted table, and sends the encrypted result back to the client interface. The client then decrypts the result using the private key and displays it to the user, ensuring that sensitive data remains secure.

References

1. Hartmann, M., Hashmi, U.S., Imran, A.: Edge computing in smart health care systems: Review, challenges, and research directions. Trans. Emerg. Telecommun. Technol. **33**(3), e3710 (2022)
2. Qu, Q., Liu, C., Bao, X.: E-commerce enterprise supply chain financing risk assessment based on linked data mining and edge computing. Mobile Inf. Syst. (2021)
3. Nain, G., Pattanaik, K.K., Sharma, G.K.: Towards edge computing in intelligent manufacturing: past, present and future. J. Manuf. Syst. **62**, 588–611 (2022)
4. El Ghmary, M., Hmimz, Y., Chanyour, T., Malki, M.O.: Energy and processing time efficiency for an optimal offloading in a mobile edge computing node. Int. J. Commun. Netw. Inf. Secur. **12**(3), 389–393 (2020)
5. Maftah, S., El Ghmary, M., El Bouabidi, H., Amnai, M., Ouacha, A.: Optimal task processing and energy consumption using intelligent offloading in mobile edge computing. Int. J. Interact. Mobile Technol. **16**(20) (2022)

6. Chen, B., Wan, J., Celesti, A., Li, D., Abbas, H., Zhang, Q.: Edge computing in iot-based manufacturing. IEEE Commun. Mag. **56**(9), 103–109 (2018)
7. Ouacha, A., Ghmary, E.: Mohamed: virtual machine migration in mec based artificial intelligence technique. IAES Int. J. Artif. Intell. **10**(1), 244 (2021)
8. El Ghmary, M., Chanyour, T., Hmimz, Y., Cherkaoui Malki, M.O.: Processing time and computing resources optimization in a mobile edge computing node. In: Embedded Systems and Artificial Intelligence, pp. 99–108. Springer (2020)
9. Hmimz, Y., El Ghmary, M., Chanyour, T., Malki, M.O.C.: Computation offloading to a mobile edge computing server with delay and energy constraints. In: 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), pp. 1–6. IEEE (2019)
10. Bak, G., Reicher, R.: Small and medium-sized enterprises' perceptions of the use of cloud services. Interdis. Descrip. Complex Syst. INDECS **21**(2), 131–140 (2023)
11. Almusallam, N., Alabdulatif, A., Alarfaj, F. et al.: Analysis of privacy-preserving edge computing and internet of things models in healthcare domain. Computat. Math. Med. (2021)
12. Cicconetti, C., Conti, M., Passarella, A.: A decentralized framework for serverless edge computing in the internet of things. IEEE Trans. Netw. Serv. Manag. **18**(2), 2166–2180 (2020)
13. Kumari, K.A., Sharma, A., Chakraborty, C., Ananyaa, M.: Preserving health care data security and privacy using carmichael's theorem-based homomorphic encryption and modified enhanced homomorphic encryption schemes in edge computing systems. Big Data **10**(1), 1–17 (2022)
14. Bahrami, M., Singhal, M.: A light-weight permutation based method for data privacy in mobile cloud computing. In: 2015 3rd IEEE International Conference on Mobile Cloud Computing, Services, and Engineering, pp. 189–198. IEEE (2015)
15. Tao, Z., Xia, Q., Hao, Z., Li, C., Ma, L., Yi, S., Li, Q.: A survey of virtual machine management in edge computing. Proc. IEEE **107**(8), 1482–1499 (2019)
16. Wang, T., Zhou, J., Chen, X., Wang, G., Liu, A., Liu, Y.: A three-layer privacy preserving cloud storage scheme based on computational intelligence in fog computing. IEEE Trans. Emerg. Topics Comput. Intell. **2**(1), 3–12 (2018)
17. AlMajed, H., AlMogren, A.: A secure and efficient ecc-based scheme for edge computing and internet of things. Sensors **20**(21), 6158 (2020)
18. Yang, M., Zhu, T., Liu, B., Xiang, Y., Zhou, W.: Machine learning differential privacy with multifunctional aggregation in a fog computing architecture. IEEE Access **6**, 17119–17129 (2018)
19. Kumar, J., Saxena, V.: Asymmetric encryption scheme to protect cloud data using paillier-cryptosystem. Int. J. Appl. Evolut. Comput. (IJAEC) **12**(2), 50–58 (2021)
20. Chandravathi, D., Lakshmi, P.V.: Privacy preserving using extended euclidean algorithm applied to rsa-homomorphic encryption technique **8**(10), 3175–3179 (2019)

Artificial Intelligence in Services and Real Problems

EDNBC: A New Efficient Distributed Naive Bayes Classifier for Vertically Distributed Data

Ahmed M. Khedr, Ibrahim Attiya, and Amal Ibrahim Al Ali

Abstract A common constraint in distributed data is that the database cannot be moved to other network sites due to computational costs, data size, or privacy considerations. All of the existing distributed Naive Bayes algorithms for classifying data are designed for horizontally distributed or special cases of vertically distributed data where different sites contain different attributes for a common set of entities. In this paper, we propose a new distributed version of the Naive Bayes Classifier (EDNBC) using a Directed Acyclic Graph (DAG) in d-dimensional space across vertically distributed databases. The main goal of the proposed version is to minimize the communication cost among the database nodes by gathering statistical summaries at each site and then aggregating these summaries to get the final results.

Keywords Distributed algorithm · Directed acyclic graph · Naive bayes classifier · Vertically distributed databases

Ahmed M. Khedr, Ibrahim Attiya and Amal Ibrahim Al Ali these authors contributed equally to this work.

A. M. Khedr
Department of Computer Science, University of Sharjah, 27272 Sharjah, UAE

A. M. Khedr (✉) · I. Attiya
Mathematics Department, Zagazig University, 44519 Zagazig, Egypt
e-mail: akhedr@sharjah.ac.ae

I. Attiya
Faculty of Computer Science and Engineering, New Mansoura University, New Mansoura City, Egypt

A. I. Al Ali
Department of Information Systems, University of Sharjah, 27272 Sharjah, UAE

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_38

1 Introduction

Data classification is a two-step process: in the first step, a model is built describing a predetermined set of data classes or concepts, and the model is constructed by analyzing database samples described by attributes. Each sample is assumed to belong to a predefined class, as determined by one of the attributes, called the class label attribute. The samples were analyzed to build the model collectively from the training data set. Since the class label of each sample is provided, this step is also known as supervised learning. In the second step, the model is used for determining the classes of newly arrived data. First, the predictive accuracy of the classifier is estimated. The accuracy of a model on a given test set is the percentage of test set samples that are correctly classified by the model. If the accuracy of the model is considered acceptable, the model can be used to classify future data of which the class label is not known. A classification technique is a systematic approach for building classification models from an input dataset.

NB classifier is widely used in Machine Learning. It can efficiently be learned, and it provides simple generative models of the data and they achieve pretty good results in various classification tasks such as text classification. Naive Bayes classifier relies on the hypothesis that the attributes of the description domain are independent conditionally to each class, i.e., conditional distributions are product distributions, but it has often been noticed that it keeps achieving good performances even when these conditions are not met [1]. Previously developed classification algorithms require all data stored in distributed locations must be transferred to a common site called the central site and recompiled as one complete and local dataset these methods are known as centralized methods. When applying one of the classification techniques, such as NB classifier algorithm, to situational input stored in geographically distributed databases, several problems can arise from the classification process. Perhaps the input data is sensitive or secure in nature, i.e., government, private business, or personal information, steps must be taken to ensure that malicious parties cannot intercept that sensitive data. These are some of the most common problems encountered when constructing a NB classifier over distributed databases using previously developed methods.

2 Related Research

The task of classification or analyzing data from a vertically distributed and geographically a distributed set of databases without moving the whole data to one place and preserve data privacy is an important task in the field of distributed databases [2–9]. A number of algorithms are designed for privacy mining horizontally distributed or special cases of vertically distributed databases where different sites contain different attributes for a common set of entities [10–16]. In [13], the authors introduced a

privacy-preserving NB classifier for horizontally distributed data and proposed a two-party protocol and a multi-party protocol to achieve it. This multiparty the protocol is built on the semi-trusted mixer model, in which each data site sends messages to two semi-trusted mixers, respectively, which run the two-party protocol and then broadcast the classification result. This model facilitates both trust management and implementation. The work in [14] provided privacy-preserving solutions for mining an NB-classifier across a database vertically distributed data mining scenario when different sites contain different attributes for a common set of entities (a special case of distributed data). Security requirements to be met in this algorithm demand that a site know only its own attributes-value pairs when they become parts of the global one. Our algorithm does not assume any conditions on the data partition at any one of the participating sites.

Justin et al. [15] presented a novel solution for NB classification over vertically distributed private data. Instead of using data transformation, they define a protocol using homomorphic encryption to exchange the data while keeping it private. In [17], the authors proposed a set of algorithms for classifying data using NB from a group of collaborative parties without breaching privacy. Privacy is also preserved using privacy preservation techniques such as offset computation and encryption. The third-party concept is also introduced to calculate global classification results with privacy preservation.

In [16], the authors presented protocols to develop a privacy-preserving NB classifier on both vertically as well as horizontally distributed data. That paper has concentrated on the semi-honest model, where each party assumed to follow the protocols but may try to infer information from the messages it sees. Vaidya et al. [16] designed for the special case of vertically distributed data where different sites contain different attributes for a common set of entities. In this paper, we propose a distributed version of NB classifier algorithm for vertically distributed data using DAG. Where DAG can reduce the communication among the participating nodes and so they exchanged messages and also can increase the security level where the communication will be done only between the parent and its children. Our proposed approach is based on a general strategy for transforming traditional machine learning algorithms into distributed learning algorithms based on the decomposition of the learning task into hypothesis generation and information extraction components; formally defined the information required for generating the hypothesis (sufficient statistics); and show how to gather the sufficient statistics from distributed data sources. Our algorithm is designed for vertically distributed databases in the most general situation in which existing distributed databases preserves the privacy and a good level of security of the data at individual sites by requiring transmission of only minimal information to other sites. In [22], Khedr et al. proposed two decomposable versions of NB classifier for horizontally and vertically distributed databases. There are a number of differences between our work and the work in [22], they assumed that the attribute-names present in each of the participating the database is known to all the other participating nodes, however, we assume that only the names of the shared attributes between parent and its children are known. The proposed the approach can be applied to

both single or multiparent nodes and it is more efficient in time and the number of exchanged messages than [22] as shown in our simulation section.

The rest of the paper is organized as follows: in Sect. 3, we describe our methodology for handling the proposed problem and the proposed algorithm, the complexity computing, and the security discussion of the proposed algorithm will be described. The study of the properties of our algorithm via simulation will be discussed in Sect. 4. We conclude our paper in Sect. 5.

3 Distributed NB Classifier

In a distributed database scenario, there are n databases located at different network nodes and all of them together constitute the global database for the global computation.

3.1 Distributed Data Integration

Nature of Data Distribution: There are two primary ways in which the databases, together, may be seen as forming an implicit global dataset D. Horizontally Distributed Database where data D exists in the form of a set of databases each of which has the same set of attributes.

Vertically Distributed Data where a dataset D exists in the form of a set of databases where each database contains a subset of the attributes of the original dataset D. In this case, each component may share some attributes with other databases.

In the situation modeled here, a DAG G(V, E) is used to represent the pattern of attribute sharing among the databases, where |V| = n representing the number of participating nodes, and |E| representing the number of edges between all participating nodes. We add an edge between two nodes when their databases share one or more attributes.

DAG provides parent/child relationships among the nodes. Here, we use a general DAG rooted at the Learner, in which some nodes may have a multi-parent node as shown in Fig. 1. We assume that there can be arbitrary overlap in the attribute sets of any pair of databases. As an abstraction, we model the database D_i at each ith site by a relation containing a number of tuples (shared values). The set of attributes contained in D_i is represented by X_i. For any pair of relations D_i and D_j the corresponding sets X_i and X_J may have a set of Shared attributes given by S_{ij}. Since an arbitrary number of independent, already existing, databases may be consulted for a computation, we cannot assume any data normalization to have been performed for their schemas. The implicit dataset D with which the computation is to be performed is a subset of the set of tuples generated by a Join operation performed on all the participating relations $D_1, D_2, \ldots, D_n$. However, the tuples of D cannot be made

explicit at any one network site by any one agent because the D_i's cannot be moved in their entirety to other network sites. The tuples of D, therefore, must remain only implicitly specified to an agent. This inability of an agent to make explicit the tuples of D is the main problem addressed in the generalized decomposition of global algorithms and is addressed in later sections.

For any parent and child nodes (databases D_i and D_j, $i \neq j$) they share some attributes, and we define the set of all those attributes that are shared among all possible pairs of databases as S_{ij}. To facilitate computations with implicitly specified sets of tuples of D, we define a set S that is the union of all the attribute intersection sets S_{ij}s, that is,

$$S = \bigcup_{i,j,i\neq j} S_{ij}, \tag{1}$$

The set S, thus, contains the names of all those attributes that are visible to more than one agent because they occur in more than one participating D_i. We define a relation Shared containing all possible enumerations for the attributes in the set S that meet at least one tuple at each participating site. This formulation of S facilitates a similar treatment for horizontally or vertically partitioned datasets because horizontal partitioning can be seen as the case where all attributes are Shared.

3.2 The Proposed Algorithm

In NB learning, each instance is described by a vector of attribute values and its class can take any value from some predefined set of values. Assume that $X_1, X_2, \ldots, X_n$ are n attributes. A set of instances with their classes, the training data, is provided. A test instance E represented by a vector $\langle x_1, x_2, \ldots, x_n \rangle$, where x_i is the value of X_i is presented. The *learner* is asked to predict its class according to the evidence provided by the training data. Let C represent the classification variable, and let c_j be the value that C takes and $c_j(E)$ to denote the class of E. Using the Bayesian

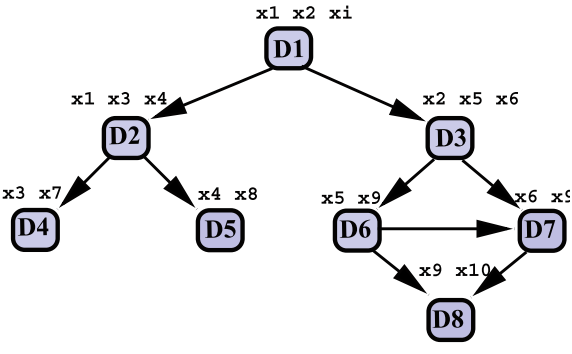


Fig. 1 Multi-parents DAG

classifier, the class of the new instance E is predicted as follows:

$$c_j(E) = \arg\max_{c_j \in C} Pr(c_j) Pr(x_1, x_2, \ldots, x_n | c_j). \quad (2)$$

Assume that the effect of an attribute value on a given class is independent of the values of the other attributes. This assumption is called class conditional independence:

$$Pr(E|c_j) = Pr(x_1, x_2, \ldots, x_n|c_j) = \prod_{i=1}^{n} Pr(x_i|c_j). \quad (3)$$

Then, the resulting classifier is called a NB classifier:

$$c_j(E) = \arg\max_{c_j \in C} Pr(c_j) \prod_{i=1}^{n} Pr(x_i|c_j). \quad (4)$$

where $c_j(E)$ is the class of the test instance E, $Pr(c_j)$ and $Pr(x_i|c_j)$ are the prior and conditional probabilities, respectively. An instance is classified into classes with maximal probabilities calculated with Eq. 4.

3.2.1 Sufficient Statistics for NB Classifier

The NB classifier involves a learning phase in which the set of probabilities $Pr(c_j)$ and $Pr(x_i|c_j)$ are estimated, based on counting the frequency of various data combinations within the training examples. The set of these estimates corresponds to the learned hypothesis. This hypothesis is then used to classify each new instance during the classification phase. The set of probabilities $Pr(c_j)$ and $Pr(x_i|c_j)$, representing the hypothesis, can be computed based on the following counts:

- N_t is the total number of training examples.
- N_{c_j} is the number of training examples in class c_j.
- $N_{x_i c_j}$ is the number of training examples in class c_j and has the attribute value x_i.

These counts represent sufficient statistics for the hypothesis built during the learning phase of the NB classifier. We will show how the minimal sufficient statistics can be computed when the data are vertically distributed.

To compute the minimal sufficient statistics, we design a distributed NB classifier. The outlines of this algorithm using DAG include three phases: The organization phase, the Learning phase, and the Classification phase. In the organization phase, the DAG construction procedure is executed to build the DAG from the geographically distributed databases. In the Learning phase, the prior and conditional probabilities are estimated based on counting the frequencies over the training examples

from the participating databases. In the Classification phase, the various terms of the estimated probabilities are then used to classify the test instance to the correct class.

3.2.2 Organization Phase

The main goal of this phase is to organize the data sources D_i's based on the pattern of attribute sharing among the databases in a DAG structure. In this phase, the DAG construction procedure is used to build the DAG from the geographically distributed databases.

3.2.3 Learning and Classification Phases

Given an instance $E = \langle x_1, x_2, \ldots, x_n \rangle$, we will compute the different counts of training examples from the partitioned data and according to these counts, we will find the predicted class of E.

distributed NB Classifier Procedure()

Step 1: Find the shared values with the parent (Top-Down strategy): Every participating node finds the shared values between the parent and its children. The shared relation will be created at the parent using these values.
Step 2: Local Computations (Bottom-Up strategy) Every participating node (from $level_{(\log n)}$ to $level_{(0)}$) will be executed in the following cases:
Case 1: if the node is a leaf node select all tuples that belong to any combination of received shared values from parent, else from the received ordered lists, select all tuples that belong to any combination of shared values and in both cases do the following:

1. From the selected tuples, for every shared value (tuple) l do
   (a) Compute the count of training examples that satisfy $cond_l$; $N(D_k)_{cond_l}$.
   (b) For every class label c_j do
      (i) If the node contains the class label attribute then compute the count of training examples in class c_j that satisfy $cond_l$; $N(D_k)_{cond_l.and.class_{cj}}$.
      (ii) Else set $N(D_k)_{cond_l.and.class_{c_j}}$ = count of training examples that satisfy $cond_l$.
      (iii) For every attribute value x_i do
         (A) Compute the count of training examples in class c_j, and having the attribute value x_i; $N(D_k)_{cond_l.and.class_{c_j}.and.attr_{x_i}}$
         (B) Create an ordered list that consists of:
            - the shared value for shared attribute between current node and its parent,
            - the count of training examples that satisfy $cond_l$,

- the corresponding class label c_j,
- the count of training examples in class c_j that satisfy $cond_l$,
- the count of training examples in class c_j, and has the attribute value x_i.

Case 2: if the node is a parent node then from all ordered lists received from children nodes and ordered lists computed from current node do:

1. For every shared value (tuple) l do
   (a) Compute the total number of training examples that satisfy $cond_l$ from the following relation:
   $$N_l = \prod_{k=1}^{n} N(D_k)_{cond_l}$$
   (b) For every class label c_j do
      (i) Compute the total number of training examples in class c_j by using the relation:
      $$N_{c_j l} = \prod_{k=1}^{n} N(D_k)_{cond_l.and.class_{c_j}}$$
      (ii) For every attribute value x_i do
         (A) Compute the total number of training examples in class c_j, and having the attribute value x_i by using the relation:
         $$N_{x_i c_j l} = \prod_{k=1}^{n} N(D_k)_{cond_l.and.class_{c_j}.and.attr_{x_i}}$$
         (B) Create an ordered list that consists of:
         - the shared value for shared attribute between current node and its parent,
         - the total number of training examples that satisfy $cond_l$,
         - the corresponding class label c_j,
         - the total number of training examples in class c_j that satisfy $cond_l$,
         - the total number of training examples in class c_j, and has the attribute value x_i.

Case 3: if the node is a Single-Parent then sends all ordered lists to parent node else (i.e., Multi-Parents) send all ordered lists to one parent and send only the shared values to other parents.

Step 3: Global Computations The following steps will be executed at *Learner* to aggregate all local counts and obtain the global counts (Totals).

1. Compute the total number of training examples by using the relation:

$$N_t = \sum_l N_l = \sum_l \left(\prod_{k=1}^{n} N(D_k)_{cond_l} \right)$$

2. Compute the total number of training examples in-class c_j by using the relation:

$$N_{c_j} = \sum_l N_{c_j l} = \sum_l \left(\prod_{k=1}^{n} N(D_k)_{cond_l .and. class_{c_j}} \right)$$

3. Compute the total number of training examples in-class c_j, and has the attribute value x_i by using the relation:

$$N_{x_i c_j} = \sum_l N_{x_i c_j l} = \sum_l \left(\prod_{k=1}^{n} N(D_k)_{cond_l .and. class_{c_j} .and. attr_{x_i}} \right)$$

4. Compute $Pr(c_j) = \frac{N_{c_j}}{N}$.
5. Compute $Pr(x_i \mid c_j) = \frac{N_{x_i c_j}}{N_{c_j}}$.
6. Classify the new instance $E = \langle x_1, x_2, \ldots, x_n \rangle$ to the class with maximal probability, $c_j(E)$ by using the relation:

$$c_j(E) = \arg\max_{c_j \in C} Pr(c_j) \prod_{i=1}^{n} Pr(x_i | c_j)$$

End Algorithm

3.3 Complexity Analysis and Security Discussion

In any instance of the global computation, we assume that one of the participating nodes is the one that needs the result of the global computation, and we mark it as the *Learner* node. A more efficient communication may be a tree-like structure with the learning node at the root. Messages flow up and down this tree and information is synthesized or inherited as it moves up or down the tree structure. This structure better preserves the locality of information and hence enhances data privacy. Also, This immediately provides an improvement in the number of messages that must be exchanged between the nodes.

3.3.1 Complexity Analysis

We show below an expression for the number of messages that need to be exchanged among stationary agents for running the NB classifier from vertically distributed data using DAG. Let n be the number of nodes in G, l be the average number of preshared (different) values of shared attribute(s) at each node, r be the average number of shared tuples at each node, and m be the number of class labels in the class label attribute (m is a very small integer).

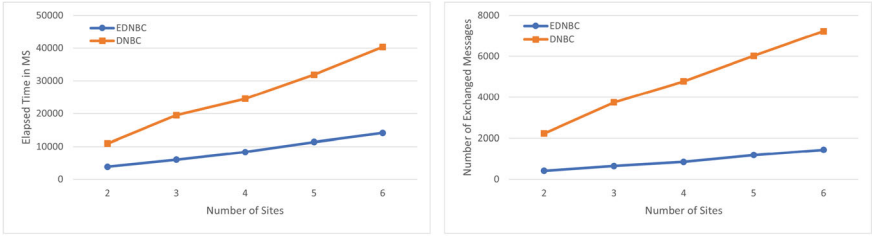
Each parent node will send all values of the shared attribute(s) to its child, so in Top-Down strategy of our algorithm $(n-1)*l$ messages will be exchanged among parents and the children. On the other hand, each child node will send shared tuple values and the corresponding local counts of training examples to its parent node, so in the Bottom-Up strategy of our algorithm $(n-1)*r*m$ messages will be exchanged between children and parents. Therefore, the total number of exchanged messages of our algorithm will be:

$$Total\ Exchanged\ Messages = (n-1)*(l+r*m). \tag{5}$$

3.3.2 Security Discussion

Now it is time to move on to the analysis of the security of our algorithm. We have successfully designed and implemented a distributed version of the NB classifier algorithm in a decentralized manner. In this decentralized set-up, we perform much more processing at each of the distributed database sites, rather than sending all of the data to a *central* location and then having that *central* server perform all of the data analysis. One can see that the results obtained from running this new algorithm are very promising. It has been shown that this methodology works very well and does indeed produce very similar results to the non-distributed method.

From the point of view of data security and privacy, there is no data tuple is exchanged between the database nodes; instead only distinct values of shared attributes need to be moved between parent and the children nodes and the value returned by a node to any parent is the output of the application of the local functions (computations). Since this is only a statistical summary and does not reveal the contents of the local database rows it is sent unaltered to the parents. If the information security and privacy is defined by not having to release any data tuple out of a database for transmission over the network and the reconstruction of any data tuple is impossible by the released data summaries then the above algorithms preserve the privacy of the data in each participating database. No data tuple is ever transmitted and the summaries are not sufficient to reconstruct any individual data tuple. Finally, our Naive Bayes classifier minimizing the information disclosure and maximizing data privacy and confidentiality.



(a) (b)

Fig. 2 **a** Elapsed time to run NB classifier when the number of local sites varied, **b** number of exchanged messages to run NB classifier when the number of local sites varied

4 Simulation Results

The tests were performed to find out the effect of various parameters on the final result. Three very important variables that affect the result are the number of tuples per database, the number of sites, and the average number of shared tuples between local databases. We have performed a number of tests to demonstrate that the NB Classifier can be computed in a distributed knowledge environment without moving all the databases to a single site. These tests have been carried out on a network of workstations connected by a LAN and tested against a number of databases of different sizes. The algorithm was implemented using Microsoft Visual C#. The databases were manipulated through Microsoft SQL Server.

The first test was done to demonstrate how the elapsed time and the number of exchanged messages vary with the number of local sites. The number of local sites varies as 2, 3, 4, 5, and 6. Figure 2a shows how the elapsed time to compute NB Classifier in an implicit database $\mathcal{D}$ changes with the number of local sites. It is clear that the elapsed time to compute NB Classifier increases as the number of local sites increases. Also, it is evident that when we use our method the elapsed time will be reduced considerably depending on the number of participating nodes. Figure 2b shows how the number of exchanged messages between the local sites changes with the number of local sites. It is evident from the figure that when utilizing our EDNBC method the number of exchanged messages is considerably reduced in comparison with the DNBC method.

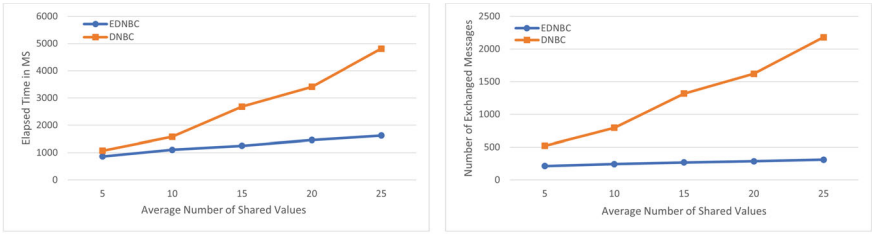
The second test was done to demonstrate how the elapsed time and the number of exchanged messages vary with the average number of shared tuples between local databases. The number of shared values varies as 5, 10, 15, 20, and 25. Figure 3a shows how the elapsed time to compute NB Classifier in an implicit database $\mathcal{D}$ changes with the average number of shared tuples between local databases. It is clear that the elapsed time to compute NB Classifier increases as the number of shared values increases. Also, it can be easily seen that when we utilize the DNBC method, the elapsed time to compute NB Classifier varies exponentially as the size of the database increases. However, when we use our EDNBC method the elapsed

time will be reduced considerably depending on the number of participating nodes. Figure 3b shows how the number of exchanged messages between the local sites changes with the number of shared values. It is evident from the figure that, when we utilize our EDNBC method the number of exchanged messages is considerably reduced in comparison with the DNBC method.

The last test aimed to illustrate how the number of exchanged messages and the elapsed time and vary with the number of tuples in the database. Figure 4a shows how the elapsed time to compute NB Classifier in an implicit database $\mathcal{D}$ varies with the number of tuples in the database. It is evident from the figure that, when using the DNBC classifier, the elapsed time to compute NB Classifier varies exponentially as the size of the database increases. However, when we use our method (EDNBC) the elapsed time will be reduced considerably. Figure 4b shows how the number of exchanged messages between the local sites varies with the number of tuples in the database. It is evident from the figure that the number of exchanged messages varies exponentially with the size of the database when using the DNBC classifier. However, when we use our method (EDNBC) the number of exchanged messages is considerably reduced depending on the total number of participating nodes.

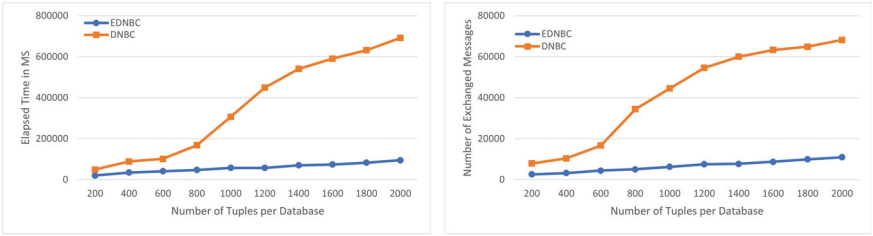
5 Conclusion

In conclusion, the NB classifier is used because of its simplicity and high efficiency. Indeed, they can efficiently be learned, they provide simple generative models of the data and they achieve pretty good results in various classification tasks. In this paper, we proposed a distributed algorithm for learning the NB classifier from vertically distributed databases that are identical to the one built from the whole database without transmitting any data tuples between sites. The proposed distributed algorithm is appropriate and efficient for distributed databases that perform global computations across geographically distributed databases by exchanging only summaries and thus preventing the transfer of any data tuples across the network. This algorithm preserves



(a) (b)

Fig. 3 **a** Elapsed time to run NB classifier when the average number of shared values varied, **b** number of exchanged messages to run NB classifier when the average number of shared values varied



(a) (b)

Fig. 4 **a** Elapsed time to run NB classifier when the number of tuples varied, **b** number of exchanged messages to run NB classifier when the number of tuples varied

the privacy and a good level of security of the data at individual sites by requiring transmission of only minimal information to other sites. Our experimental results have shown that the proposed method works very well and does indeed produce very similar results to the non-distributed (centralized) method. Also, it shows that there is a significant reduction in the amount of disclosed information, communications, and computational costs compared with existing algorithms.

References

1. Domingos, P., Pazzani, M.: On the optimality of the simple Bayesian classifier under zero-one loss. Mach. Learn. **29**, 103–130 (1997)
2. Khedr, A.M.: Decomposable algorithm for computing k-nearest neighbors across partitioned data. Int. J. Parallel, Emerg. Distrib. Syst. **31**(4), 334–353 (2016). https://doi.org/10.1080/17445760.2015.1057820
3. Khedr, A.M., Al Aghbari, Z., Ali, A.A., Eljamil, M.: An efficient association rule mining from distributed medical databases for predicting heart diseases. EEE Access **9**, 15320–15333 (2021)
4. Khedr, A.M.: EDCP: effective decomposable closest pair algorithm for distributed databases. Eng. Lett. **28**(3), 930–938 (2020)
5. Khedr, A.M., Osamy, W., Salim, A., Abbas, S.: A novel association rule-based data mining approach for internet of things based wireless sensor networks. EEE Access **8**, 151574–151588 (2020)
6. Khedr, A.M., Osamy, W., Salim, A., Salem, A.: Privacy-preserving data mining approach for IoT based WSN in smart city. Int. J. Adv. Comput. Sci. Appl. **10**(8), 555–563 (2019)
7. Khedr, A.M., Al Aghbari, Z., Kamel, I.: Privacy preserving decomposable mining association rules on distributed data. Int. J. Eng. Technol. **7**(313), 157–164 (2018)
8. Khedr, A.M., Bhatnagar, R.: New algorithm for clustering distributed data using K-means. Comput. Inform. **33**(4), 943–964 (2014)
9. Khedr, A.M., Pravija Raj, P.V.: DRNNA: decomposable reverse nearest neighbor algorithm for vertically distributed databases. In: 2021 18th International Multi-Conference on Systems, Signals & devices (SSD), pp. 681–686 (2021)
10. Lindell, Y., Pinkas, B.: Privacy-preserving data mining. In: Advances in Cryptology, Lecture Notes in Computer Science, vol. 1880, pp. 36–53. Springer-Verlag (2000)
11. Vaidya, J., Clifton, C.: Privacy preserving association rule mining in vertically partitioned data. In: Proceedings of the 8th ACM International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, pp. 639–644 (2002)

12. Kantarcioglu, M., Vaidya, J.: Privacy-preserving Naive Bayes classifier for horizontally partitioned data. In: IEEE Workshop on Privacy Preserving Data Mining, pp. 3–9 (2003)
13. Xun, Y., Yanchun, Z.: Privacy-preserving Naive Bayes classification on distributed data via semi-trusted mixers. Inform. Syst. **34**(3), 371–380 (2009)
14. Vaidya, J., Clifton, C.: Privacy Preserving Naive Bayes classifier for vertically partitioned data. In: IEEE Workshop on Privacy Preserving Data Mining, pp. 3–9 (2003)
15. Justin, Z., Stan, M., Liu, C.: Privacy-preserving Naive Bayesian classification over vertically partitioned data. In: Proceedings of the Fifth International Conference on Electronic Business, Hong Kong, pp. 483–488 (2005)
16. Vaidya, J., Kantarcioglu, M., Clifton, C.: Privacy-preserving ISBN: 978-0-9891305-4-7 2014 SDIWC 48 Naive Bayes classification. Int. J. Very Large Data Bases **17**(4), 879–898 (2008)
17. Keshavamurthy, B.N., Toshniwal, D.: Privacy-Preserving Naive Bayes Classification Using Trusted Third Party Computation over Distributed Progressive Databases, Vol. 131, pp. 24–32. Springer Verlag Berlin Heidelberg (2011)
18. Khedr, A.M.: Learning k-nearest neighbors classifier from distributed data. Comput. Inform. **27**, 355–376 (2008)
19. Khedr, A.M., Bhatnagar, R.: A Decomposable Algorithm for Minimum Spanning Tree. Distributed Computing-Lecture Notes in Computer Science, vol. 2918, pp. 33–44. Springer-Verlag, Heidelberg (2004)
20. Khedr, A.M., Bhatnagar, R.: Agents for integrating distributed data for complex computations. Comput. Inform. **26**(2), 149–170 (2007)
21. Khedr, A.M., Salim, A.: Decomposable algorithms for nearest neighbor computing. J. Parallel Distrib. Comput. **68**(7), 902–912 (2008)
22. Khedr, A.M.: Decomposable Naive Bayes classifier for partitioned data. J. Comput. Inform. **31**(6), 1511–1531 (2012)

Discrete Reptile Search Algorithm-Based Clustering Technique for Flying Ad Hoc Networks

P. V. Pravija Raj, Ahmed M. Khedr, and Reham R. Mostafa

Abstract Given the continually evolving applications of Flying Ad Hoc Networks (FANETs), sophisticated clustering methods become more crucial for preserving network stability despite the dynamic flight characteristics of Unmanned Aerial Vehicles (UAVs). This guarantees stable communication for the seamless exchange of critical data and collaboration within FANETs. Motivated by this, we introduce a novel Discrete Reptile Search Algorithm-Based Clustering method (DRSAC) for FANETs. The Reptile Search Algorithm (RSA) is a recent meta-heuristic optimizer that yields superior results in a variety of optimization problems, with properties like minimal parameter tweaks, strong optimization stability, and ease of implementation. DRSAC efficiently organizes nodes into clusters by employing the discrete version of RSA, enhancing communication efficiency and adaptability in dynamic airborne environments. It serves as a robust solution for cluster formation and maintenance by avoiding frequent cluster reconfigurations. The best count of clusters in the FANET is decided by taking into account the constraints related to network bandwidth and node coverage. By incorporating a novel mechanism for determining more stable and efficient Cluster Heads (CHs), DRSAC extends the cluster lifetime, decreases latency, and maximizes the data delivery. This research contributes to advancing the reliability and effectiveness of FANETs, positioning DRSAC as an effective clustering solution for FANET applications. The simulation results demonstrate that DRSAC

P. V. Pravija Raj · A. M. Khedr (✉)
Department of Computer Science, University of Sharjah, Sharjah 27272, UAE
e-mail: akhedr@sharjah.ac.ae

P. V. Pravija Raj
e-mail: p20230903@dubai.bits-pilani.ac.in

P. V. Pravija Raj
Department of Computer Science, BITS Pilani, Dubai Campus, Dubai, UAE

R. R. Mostafa
Big Data Mining and Multimedia Research Group, CDAC, RISE, University of Sharjah, Sharjah 27272, UAE
e-mail: reldeiasti@sharjah.ac.ae

Information Systems Department, Faculty of Computers and Information Sciences, Mansoura University, Mansoura 35516, Egypt

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_39

exhibits efficient performance across various metrics, including cluster lifetime, data transmission rate, and energy efficiency.

1 Introduction

Flying Ad-hoc Networks (FANETs) are emerging as a major domain offering a promising platform for innovative applications across various sectors [1, 2]. While FANETs exhibit reliability, scalability, and endurance, they also introduce novel challenges to Unmanned Aerial Vehicle (UAV) communication and connectivity capabilities [3, 4]. The dynamic nature of UAV movement, coupled with 3D mobility and resource constraints, poses significant hindrances to achieving reliable communication within FANETs [5]. Overcoming these challenges necessitates the implementation of effective clustering mechanisms capable of adapting to dynamic changes and ensuring dependable communication, thereby extending UAV endurance and mission duration [6–11]. Given the critical role of Cluster Head (CH) assignment and cluster formation, the development of a robust clustering scheme is crucial to enhancing FANET performance [12–14]. Various studies have proposed a range of clustering techniques for FANETs, integrating both traditional and meta-heuristic methodologies [15–18]. The limited consideration of FANETs' distinctive characteristics in previous research has left gaps in handling the continually evolving FANET structure and improving performance. Considering the various constraints inherent in this field, this research presents a new method called the Discrete Reptile Search Algorithm-Based Clustering method (DRSAC) for FANET. Drawing inspiration from the Reptile Search Algorithm (RSA) [19], a recent meta-heuristic optimizer, distinguished by its unique search strategies and superior performance relative to other optimization techniques, DRSAC is designed to address the aforementioned challenges. By employing the discrete version of RSA, DRSAC proficiently organizes nodes into clusters, thereby augmenting communication efficiency and adaptability in dynamic airborne environments, all while mitigating the need for frequent cluster reconfiguration. DRSAC facilitates the identification of stable CHs, contributing to the reduction of energy consumption, extension of cluster lifetime, and maximization of packet delivery rates.

The key contributions of this study include:

1. We develop the DRSAC clustering technique for FANET-based applications that selects efficient and stable CHs based on multiple parameters.
2. We formulate an effective fitness function to facilitate the best selection of CHs within FANET.
3. To enhance decision-making in CH selection, we adapt and utilize the RSA, a recent and powerful Swarm Intelligent (SI) optimization approach.
4. We conduct an in-depth experimental study to reveal the effectiveness of DRSAC across various metrics, such as packet delivery rate, energy usage, and cluster lifespan.

The rest of the sections follow this structure: The related research is covered in Sect. 2, and the system model is provided in Sect. 3. In Sect. 4, the DRSAC technique is explained. The results are presented in Sect. 5. The work is concluded in Sect. 6.

2 Related Works

A clustering technique utilizing Moth Flame Optimizer (MFO) is proposed in [15], based on energy and location details of nodes. However, it ignores the mobility aspect and applies the method in a 2D environment. A Glowworm Swarm Optimization (GSO) based clustering is presented by [20] considering expended energy and location details of UAVs. A combination of Krill-Herd and GSO is devised in [14] for clustering to guarantee efficient routing and enhanced energy efficiency. However, continuous topological changes shorten the cluster lifespan, affecting overall network performance. FANET clustering using MFO is presented in [18], where CHs are chosen according to k-means sorted fitness. However, the cluster stability is affected because the UAV velocity and mobility factors are overlooked. In [24], the preferred clusters count in FANET is determined using the k-means++ method and establishes them initially. Subsequently, a weighted sum strategy is used to choose the CHs. However, this approach has a relatively short cluster lifetime. On the other hand, [21] adopted a density-based method with k-means to choose CHs, leading to extended CH lifespan and reduced overhead. However, this approach overlooks the mobility aspects and connection stability among UAVs during the clustering phase. Furthermore, its application in a 2D environment fails to align with the distinctive 3D nature of FANETs. In [22], the k-means algorithm is used to create clusters. While the method takes into account parameters like cluster size and CH coverage, it's important to note that the resulting clusters need ongoing maintenance due to less stability. [23] also used the k-means method to solve FANET clustering issues. However, the ineffectual selection of k value leads to a short cluster lifespan. In [16], another swarm-based approach is put forward. The skyline operator is utilized to improve routing performance and speed up the search process, and the Sparrow Search Algorithm (SSA) is used to create initial clusters.

As evident from the aforementioned details, many of the approaches overlooked the rapidly evolving dynamic aspects of FANETs, which could lead to solutions lacking real-time adaptability. The insufficient attention to resource constraints and dynamic behaviors in current literature may make proposed algorithms impractical for resource-constrained UAVs. Ineffective CH selection can cause delays, data loss, unequal coverage, and unbalanced node loads, which can shorten the lifespan of FANETs and cause early energy depletion of nodes. DRSAC is presented as a solution to address these issues and accomplish effective clustering in FANETs.

3 System Model

The DRSAC system model, depicted in Fig. 1, encompasses a network of N UAVs distributed throughout a 3D space, represented symbolically as an undirected graph $G_N = (V_N, E_N)$. Here, V_N denotes the UAVs and E_N denotes the connections within the graph. Each UAV can communicate wirelessly and is aware of its location. The UAVs are structured to form clusters, each consisting of several CMs supervised by a CH. (UAV_{ID}) is the unique identifier given to each UAV. The inter-distance between UAVs changes over time as they move around the network. UAV communication is facilitated through the exchange of HELLO messages (MSG_{Hello}), containing essential fields like (U_{ID}), velocity (U_{vx}, U_{vy}, U_{vz}), position (U_x, U_y, U_z), and direction. Neighbor UAVs communicate when their distance is less than the communication range R. Therefore, for any pair of UAVs $(U1, U2)$ in V_N, if their Euclidean distance $(dist(U1, U2))$ is less than R, the pair $U1, U2$ forms an edge in E_N.

UAVs utilize energy for flight, hovering, and data relay. The energy needed to send w bits of information across a distance dis can be stated mathematically as below:

$$E_{TRD}(w, dis) = \begin{cases} w * E_{el} + w * \epsilon_f * dis^2, & \text{if dis } < dt \\ w * E_{el} + w * \epsilon_a * dis^4, & \text{otherwise} \end{cases} \tag{1}$$

where E_{el} is the electronic energy usage parameter and the threshold distance is dt. The amplifier energy elements are defined by the values ϵ_f and ϵ_a.

The energy for reception of w-bits message is given by:

$$E_{RCD}(w) = w * E_{ele} \tag{2}$$

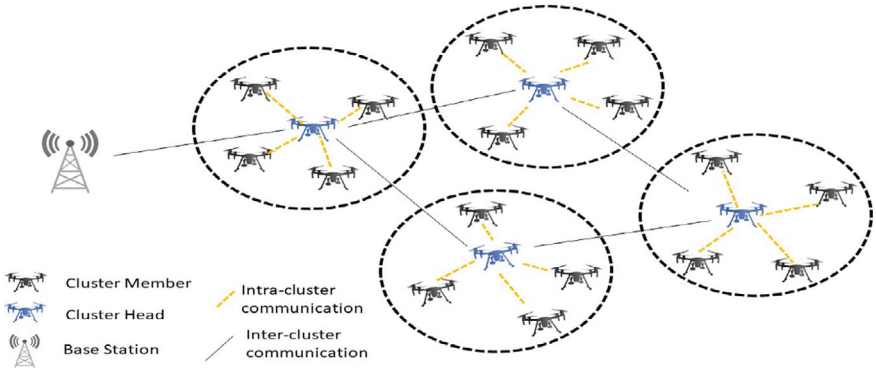


Fig. 1 Proposed system model

The hovering and flying power usage in FANET is given by:

$$P_{U_h} = \sqrt{\frac{(m_U \cdot g)^3}{2 \cdot \pi \cdot w_r \cdot w_n \cdot \rho_a}} \tag{3}$$

$$P_{U_f} = (P_{\max} - P_{U_h}) \left[\frac{v_U(t)}{v_{\max}} \right] \tag{4}$$

where $v_{\max}$ denotes the maximum speed of flight of UAV, m_U denotes its mass, and $v_U(t)$ denotes the UAV's flying speed at t. ρ_a is the air density parameter, and g is the acceleration due to gravity. w_n and w_r stand for the wings count and radius, respectively. The hovering energy usage is given by:

$$E_{U_h} = P_{U_h} t_h \tag{5}$$

The flying energy usage is given by:

$$E_{U_f} = \int_0^{t_f} P_{U_f} \, dt \tag{6}$$

where t_h and t_f denote the hovering and flying times of UAV.

4 Proposed DRSAC Method

DRSAC utilizes a discrete version of RSA [19] to efficiently cluster nodes, improving communication and adaptability in changing airborne environments while reducing cluster reconfiguration. The $SigF$ function, known for its effectiveness in solving discrete optimization problems, is employed for discretization as follows:

$$B_a = \begin{cases} 1, & \text{if SigF(a)} > \zeta \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where, ζ is a random number between 0 and 1, specifically set to 0.5, and $SigF(a) = \frac{1}{1+e^{-a}}$ denotes the sigmoid function. Each index corresponds to a specific UAV node labeled as UAV_{ID}. A value of 1 at an index indicates that the corresponding UAV serves as a CH, and a value of 0 denotes that it is not a CH. This representation simplifies indicating whether a UAV functions as a CH in the solution.

The pseudocode for the DRSAC method is provided by Algorithm 1.

The fitness function is created for CH selection using the important components, which include (i) residual energy, (ii) intra-cluster, and (iii) BS distances, (iv) cluster load, and (v) mobility. These normalized characteristics are given certain weights and included in the formulated fitness function. Equation 8 is used to estimate the fitness

Algorithm 1 Proposed Clustering Algorithm

Input: UAVs $Ui(i = 1, 2, \hat{a}\L!, N)$, T (maximum iterations), fitness function f_{CH}.
Output: best clustering solution with $CHs = CH_k \in C$ $(k = 1, 2, ..., m)$
BEGIN
RSA parameter initialization and random initial population generation.
while iteration $t < T$ **do**
  Discretize the population based on Equation 7,
  Check and fix if there is any infeasible individual.
  Update the RSA parameters [19].
  for each crocodile (C_i) in the population **do**
    Identify CHs, CHs = list of C indices with value equals 1.
    Generate clusters, fix any infeasibility, and update CHs and clusters.
    Evaluate each solution based on $f_{CH}()$ given by Equation 8.
    Find the Best solution so far.
    Compute the new crocodile location C_i in terms of t value [19].
  end for
  Increment t, $t = t + 1$.
end while
Return the best clustering solution.
END

value. The allocated weights are represented by $\gamma_i (i = 1, 2, ..., 5)$, with the sum equal to 1. The choice of weight values is based on the user-specified requirements of the real-world application scenario.

$$f_{CH} = \gamma_1 \cdot f_n(en) + \gamma_2 \cdot f_n(din) + \gamma_3 \cdot f_n(dbs) + \gamma_4 \cdot f_n(dmp) + \gamma_5 \cdot f_n(lo) \tag{8}$$

where,

– The residual energy function, denoted by $f_n(en)$, is given by dividing the total residual energy of all UAVs by the total residual energy of all CHs.

$$f_n(en) = \frac{\sum_{i=1}^{N} E(Ui)}{\sum_{k=1}^{m} E(CH_k)} \tag{9}$$

where the residual energy of the ith UAV and the kth CH in the FANET are shown by the symbols $E(Ui)$ and $E(CH_k)$, respectively.

– The intra-cluster distance in the FANET is given by the function $f_n(din)$,

$$f_n(din) = \frac{\sum_{k=1}^{m} \sum_{l=1}^{Mk} Dis(CH_k, M_{l,k})}{\sum_{i=1}^{N} \sum_{j=1}^{n(Ui)} Dis(Ui, Uj)} \tag{10}$$

where $Dis(CH_k, M_{l,k})$ indicates the CH to member distances, while $Dis(Ui, Uj)$ denotes the neighboring UAV distances. The lower the value of $f_n(din)$, the less communication energy is consumed by that specific set of CHs.

– $f_n(dbs)$ refers to the CHs to BS distances. The chosen group of UAVs to function as CHs performs better when $f_n(dbs)$ is lower.

$$f_n(dbs) = \frac{\sum_{k=1}^{m} Dis(CH_k, BS)}{\sum_{i=1}^{N} Dis(Ui, BS)} \tag{11}$$

– The mobility values of the CHs relative to other nodes in terms of speed and position values is given by $f_n(dmp)$:

$$f_n(dmp) = \frac{\sum_{k=1}^{m} MP_{CH_k}}{\sum_{l=1}^{N} MP_{Ul}} \tag{12}$$

where MP_{Ul} and MP_{CH_k} denote the average speed deviation with respect to its neighbors, respectively.

– The load balancing factor, denoted as $f_n(lo)$, is calculated as the overall degree of variations between the average cluster size and the individual cluster sizes $(Csz_k, k = 1, 2, ..., m)$.

$$f_n(lo) = \sum_{k=1}^{m} \left| \frac{N}{m} - Csz_k \right| \tag{13}$$

4.1 Complexity Analysis

The time complexity of the DRSAC is estimated as follows: RSA is adapted and utilized in DRSAC for the clustering process. Assuming a population size p, a dimension n, and maximum iterations t, the time complexity of RSA is $O(p \times (t \times n + 1))$ [19]. In addition to basic RSA, DRSAC includes discretization, a check for feasibility, and a fix step for the generated population. The complexity of DRSAC is similar to RSA and can be simplified to $O(p \times (t \times n + 1))$. Therefore, the overall time complexity of DRSAC is $O(p \times (t \times n + 1))$.

5 Simulation Results

In this section, we assess the performance of DRSAC using various metrics. MATLAB is employed to conduct simulation experiments. Table 1 provides the settings employed in the simulation. Figure 2 depicts the variation in the number of CHs generated using the DRSAC method across different numbers of UAVs. The dynamic character of the network is reflected in the corresponding rise in CHs with the increase in the number of UAVs. Remarkably, the DRSAC method excels in selecting CHs, as evidenced by the results that follow. The DRSAC approach reduces the likelihood of clusters with isolated or few nodes by employing an efficient clustering procedure, improving the overall cluster structure.

Table 1 Simulation setting

Parameters	Values
Network size	2000 m × 2000 m × 500 m
UAV transmission range	300 m
Minimum distance between UAVs	8 m
Speed of UAV	10–30 m/s
Simulation time	160 s
Mobility	RPGM-Reference point mobility model
UAVs count	20–100
Carrier frequency (Intra-cluster)	2.4 GHz
Air density	1.23 kg/m^3
Carrier frequency (Inter-cluster)	5 GHZ
E_{el}	50 nJ/bit
Acceleration due to gravity	9.8 m/s
ϵ_a	0.01 pJ/bit/m^4
Packet size	512 bytes and CBR (2 Mbps)
ϵ_f	100 pJ/bit/m^2

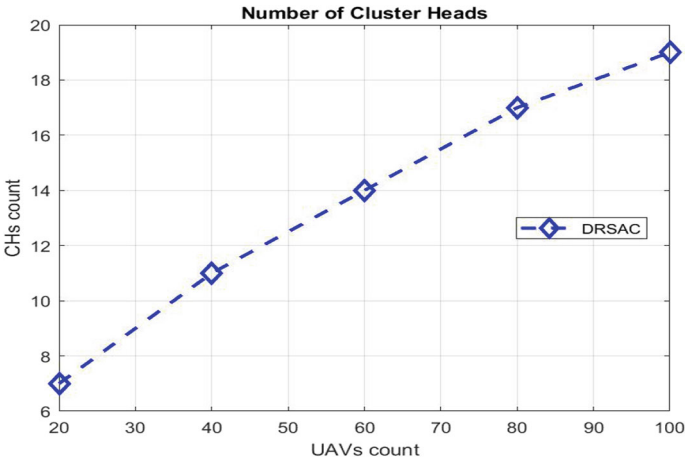


Fig. 2 CHs count results for DRSAC

Figure 3 gives the cluster creation time. The reduced cluster creation time is attributed to the faster and more precise convergence rate facilitated by the integration of discrete RSA in DRSAC. This optimized selection of CHs and the resulting cluster structure effectively balances overall energy consumption, thus prolonging the lifespan of FANET.

The average cluster lifetime results are shown in Fig. 4, which indicates a tendency for the cluster lifetime to decrease with the number of UAVs. This is because UAVs entering and departing clusters cause the FANET topology to become dynamic.

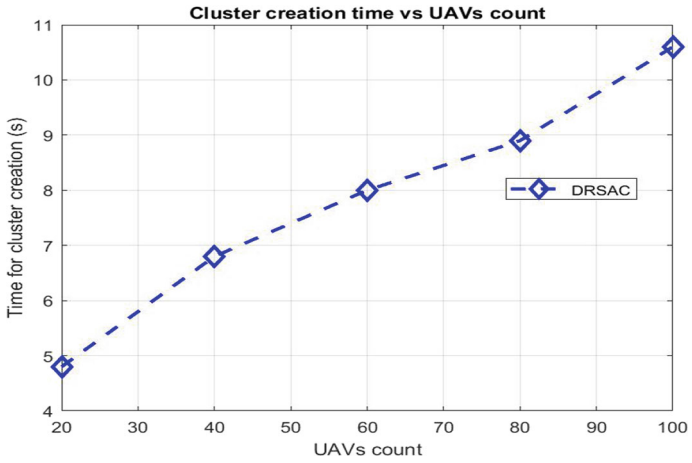


Fig. 3 Cluster creation time results for DRSAC

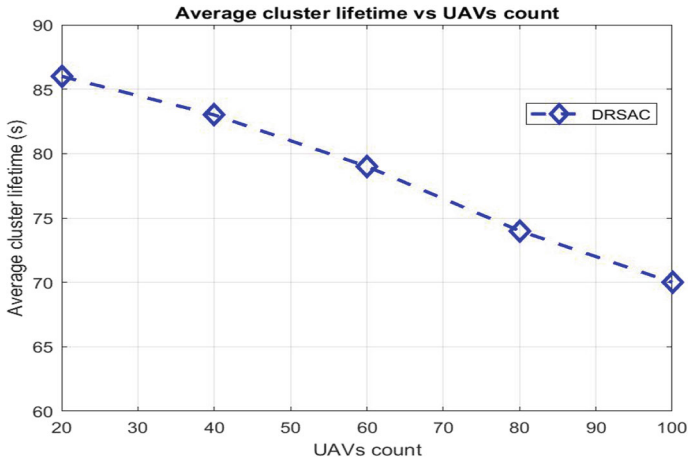


Fig. 4 Cluster lifetime results for DRSAC

Maintaining the communication reliability of FANETs is critical, especially while handling periodic topological changes. DRSAC's stable clustering strategy, which takes into account various critical parameters, has reduced the frequency of CH changes and delivered more reasonable cluster lifetime results.

Figure 5 illustrates the energy efficiency of DRSAC, which can be attributed to its efficacious clustering procedure and well-chosen CHs. Furthermore, the improved transmission reliability eliminates the requirement for node retransmissions, saving bandwidth and decreasing expended energy costs. The Packet Delivery Rate (PDR) result for different numbers of UAVs is shown in Fig. 6. The result makes it clear that DRSAC performs well in terms of packet delivery. By forming stable clusters,

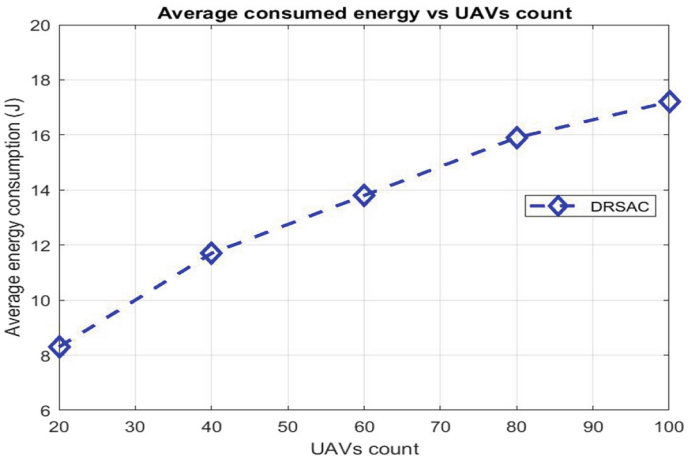


Fig. 5 Energy consumption results for DRSAC

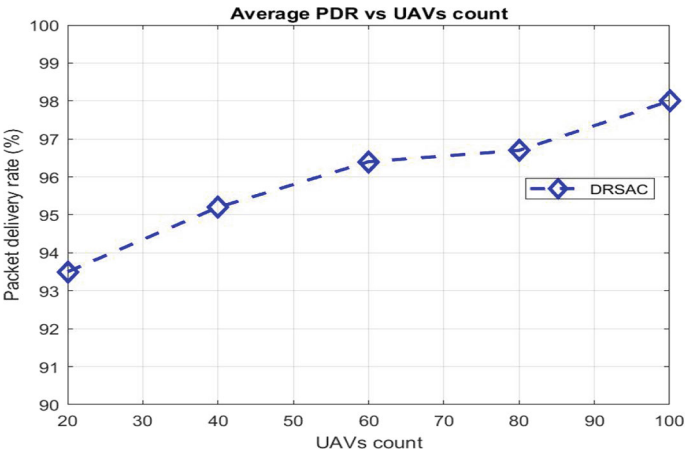


Fig. 6 Packet delivery rate results for DRSAC

DRSAC improves PDR, lowers expended energy, and prolongs network lifetime by reducing the need for frequent maintenance of clusters in FANET.

6 Conclusion

Considering that the UAVs exhibit dynamic behavior, designing an efficient clustering strategy for FANET is a difficult task. As a result, traditional methods for clustering are unable to be applied directly to FANETs. The DRSAC technique is proposed in this study to improve FANET performance under various impediments.

In order to choose CHs, a novel clustering algorithm based on discrete RSA is put forth and utilized, taking into account a number of critical aspects like power usage, distance, mobility, and load balancing. The results of the simulation show the effectiveness of DRSAC in enhancing cluster lifespan and stability, utilizing energy more effectively, and improving data delivery. The future work involves conducting an in-depth comparative analysis between DRSAC and existing research, as well as adding enhancements to further improve its performance.

Declarations

(i) Funding: N.A.
(ii) Conflicts of interest: N.A.
(iii) Ethical Approval: N/A.
(iv) Availability of data and material: N/A.
(v) Authors' contributions: The idea and design of the study were contributed to by all authors equally. Each author has reviewed and approved the final version of the document.

References

1. Chriki, A., Touati, H., Snoussi, H., Kamoun, F.: FANET: communication, mobility models and security issues. Comput. Netw. **163**, 106877 (2019)
2. Yu, S., Das, A.K., Park, Y., Lorenz, P.: SLAP-IoD: secure and lightweight authentication protocol using physical unclonable functions for internet of drones in smart city environments. IEEE Trans. Veh. Technol. **71**(10), 10374–10388 (2022)
3. Agrawal, J., Kapoor, M., Tomar, R.: A ferry mobility based direction and time-aware greedy delay-tolerant routing (FM-DT-GDR) protocol for sparse flying ad-hoc network. Trans. Emerg. Telecommun. Technol. **33**(9), e4533 (2022)
4. Lakew, D.S., Sa'ad, U., Dao, N.N., Na, W., Cho, S.: Routing in flying ad hoc networks: a comprehensive survey. IEEE Commun. Surv. Tutor. **22**(2), 1071–1120 (2020)
5. Cui, Y., Zhang, Q., Feng, Z., Wei, Z., Shi, C., Yang, H.: Topology-aware resilient routing protocol for FANETs: an adaptive Q-learning approach. IEEE Internet Things J. **9**(19), 18632–18649 (2022)
6. Pravija Raj, P.V., Khedr, A.M., Al Aghbari, Z.: EDGO: UAV-based effective data gathering scheme for wireless sensor networks with obstacles. Wirel. Netw. **28**(6), 2499–2518 (2022)
7. Raj, P.P., Khedr, A.M., Aghbari, Z.A.: An enhanced evolutionary scheme for obstacle-aware data gathering in UAV-assisted WSNs. J. Ambient. Intell. Hum.Ized Comput., 1–13 (2022)
8. Pravija Raj, P.V., Al Aghbari, Z., Khedr, A.M.: ETP-CED: efficient trajectory planning method for coverage-enhanced data collection in WSN. Wirel. Netw., 1–16 (2023)
9. Wheeb, A.H.: Flying Ad hoc Networks (FANET): performance evaluation of topology based routing protocols. Int. J. Interact. Mob. Technol. **16**(4), 137–149 (2022)
10. Orozco-Lugo, A.G., McLernon, D.C., Lara, M., Zaidi, S.A.R., González, B.J., Illescas, O., Rodríguez-Vázquez, R.: Monitoring of water quality in a shrimp farm using a FANET. Internet Things **18**, 100170 (2022)

11. Joshi, A., Dhongdi, S., Kumar, S., Anupama, K.R.: Simulation of multi-UAV ad-hoc network for disaster monitoring applications. In: 2020 International Conference on Information Networking (ICOIN), pp. 690–695. IEEE (Jan 2020)
12. Khedr, A.M., Al Aghbari, Z., Raj, P.P.: An enhanced sparrow search based adaptive and robust data gathering scheme for WSNs. IEEE Sens. J. **22**(11), 10602–10612 (2022)
13. Lee, S.W., Ali, S., Yousefpoor, M.S., Yousefpoor, E., Lalbakhsh, P., Javaheri, D., Hosseinzadeh, M.: An energy-aware and predictive fuzzy logic-based routing scheme in flying ad hoc networks (FANETs). IEEE Access **9**, 129977–130005 (2021)
14. Khan, A., Aftab, F., Zhang, Z.: BICSF: bio-inspired clustering scheme for FANETs. IEEE Access **7**, 31446–31456 (2019)
15. Khan, A., Khan, S., Fazal, A.S., Zhang, Z., Abuassba, A.O.: Intelligent cluster routing scheme for flying ad hoc networks. Sci. China Inf. Sci. **64**(8), 182305 (2021)
16. Khedr, A.M., Salim, A., PV, P. R., & Osamy, W.: MWCRSF: Mobility-based weighted cluster routing scheme for FANETs. Veh. Commun. **41**, 100603 (2023)
17. Daneshvar, S.M.H., Mohajer, P.A.A., Mazinani, S.M.: Energy-efficient routing in WSN: a centralized cluster-based approach via grey wolf optimizer. IEEE Access **7**, 170019–170031 (2019)
18. Bharany, S., Sharma, S., Bhatia, S., Rahmani, M.K.I., Shuaib, M., Lashari, S.A.: Energy efficient clustering protocol for FANETs using moth flame optimization. Sustainability **14**(10), 6159 (2022)
19. Abualigah, L., Abd Elaziz, M., Sumari, P., Geem, Z.W., Gandomi, A.H.: Reptile Search Algorithm (RSA): a nature-inspired meta-heuristic optimizer. Expert. Syst. Appl. **191**, 116158 (2022)
20. Khan, A., Aftab, F., Zhang, Z.: Self-organization based clustering scheme for FANETs using Glowworm Swarm Optimization. Phys. Commun. **36**, 100769 (2019)
21. Aadil, F., Raza, A., Khan, M.F., Maqsood, M., Mehmood, I., Rho, S.: Energy aware cluster-based routing in flying ad-hoc networks. Sensors **18**(5), 1413 (2018)
22. Bhandari, S., Wang, X., Lee, R.: Mobility and location-aware stable clustering scheme for UAV networks. IEEE Access **8**, 106364–106372 (2020)
23. Raza, A., Khan, M.F., Maqsood, M., Haider, B., Aadil, F.: Adaptive k-means clustering for Flying Ad-hoc Networks. KSII Trans. Internet Inf. Syst. (TIIS) **14**(6), 2670–2685 (2020)
24. Yang, X., Yu, T., Chen, Z., Yang, J., Hu, J., Wu, Y.: An improved weighted and location-based clustering scheme for Flying Ad Hoc Networks. Sensors **22**(9), 3236 (2022)

Data Quality Assessment of a Utility Company's Geographic Information System

Souhaila Akrikez, Mohammed Ammari, and Abdellah Idrissi

Abstract In the rapidly evolving landscape of digital transformation, the reliability of systems and databases hinges crucially on effective data quality assessment. Whether derived from simple SQL queries or complex machine learning models, the decisions made based on this data greatly depend on its quality. GIS, or geographic information systems, are a case in point: they enable the gathering, storage, analysis, and visualization of geographic data. GIS serves as a crucial tool for informed decision-making in a range of industries where decisions are based on geographic information, from urban planning and agriculture to logistics and emergency response. This paper delves into a real-world examination of data quality assessment within the geographic information system of a Moroccan utility company, Redal, offering insights and practical recommendations for enhancement.

1 Introduction

Redal is a prominent utility company in Morocco, dedicated to distributing drinking water, managing wastewater, and providing electricity services across various regions including Rabat-Sale prefecture, Temara, Skhirat, Bouznika, and Cherrat communes. At the core of Redal's operations lies its Geographic Information System (GIS), a pivotal tool that serves as the foundational repository for both spatial and non-spatial data pertaining to the infrastructure across three networks. The GIS meticulously catalogs every installed asset within the water, sanitation, and electricity networks across Redal's jurisdiction, facilitating swift and efficient location as required.

S. Akrikez (✉) · M. Ammari · A. Idrissi
Artificial Intelligence and Data Science Group, IPSS Team, Computer Science Laboratory, Computer Science Department, Faculty of Science, Mohammed V University, Rabat, Morocco
e-mail: souhaila.akrikez@um5r.ac.ma

M. Ammari
e-mail: m.ammari@um5r.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_40

The GIS serves as the linchpin for Redal's decision-making processes, exerting considerable influence over the company's operational efficiency and strategic planning. Various ancillary systems, including SIGID and VAMS, rely heavily on the GIS for their functionality and effectiveness. SIGID, aimed at optimizing field team mobility and troubleshooting interventions, and VAMS, designed for asset tracking and preventive maintenance planning, both leverage the GIS as their cornerstone. By aligning GIS data with SIGID records, Redal can forecast potential equipment failures, thereby preemptively addressing maintenance needs.

Furthermore, the GIS empowers Redal to conduct comprehensive analyses based on topological and data relationships. These analyses range from tracing power lines to identify customers affected by outages to isolating pipelines constructed from specific materials or installed before certain dates. Such capabilities enable Redal to swiftly respond to service disruptions and strategically plan infrastructure maintenance and replacement projects. For instance, GIS applications enable the identification of trends in main breaks, allowing the company to prioritize rehabilitation efforts based on criteria such as pipe material, diameter, age, and soil type, while also coordinating with other utility projects [30].

Despite the critical role played by Redal's GIS in decision-making and planning, the effectiveness of the system hinges on the quality of its underlying data. High-quality data is indispensable for ensuring the reliability and validity of GIS outputs, thereby enhancing the company's operational resilience and service delivery capabilities. This paper endeavors to assess the data quality within Redal's GIS infrastructure and propose measures for enhancing its effectiveness and reliability. Through a meticulous evaluation of data quality parameters, this study aims to identify areas for improvement, ultimately bolstering Redal's ability to make informed decisions and deliver high-quality services to its customers.

2 Literature Review

The literature review on data quality in Geographic Information Systems (GIS) draws upon four pivotal studies: [5, 24, 28, 34]. These studies collectively offer a comprehensive overview of the dimensions, methodologies, and importance of data quality in GIS, particularly focusing on participatory GIS research, the rising concern for geospatial data quality, and various assessment methods.

Key Dimensions and Types of Data Quality [5]: Batini's study delineates several critical dimensions of data quality, including accuracy, completeness, consistency, timeliness, inherent dimensions, schema dimensions, and data dimensions. These dimensions serve as foundational criteria for assessing and improving data quality, ensuring that data is reliable, current, and representative of real-world values. The study also outlines strategies and techniques for data quality improvement, such as data-driven and process-driven strategies, acquisition of new data, standardization, and the application of algorithms, heuristics, and knowledge-based activities.

Contributions to Participatory GIS Research [28]: Musungu's work enhances participatory GIS by assessing spatial data quality within informal settlements in Cape Town, emphasizing the application of traditional GIS data quality criteria to participatory methods. This study bridges a gap in literature by demonstrating the applicability and importance of rigorous data quality assessments in participatory GIS projects, thereby bolstering the credibility and utility of spatial data collected by local communities.

Growing Concern for Geospatial Data Quality [34]: VEREGIN's study discusses the heightened concern for geospatial data quality in recent years, attributing it to the expanding role of GIS in decision-making processes. Poor quality data can lead to inaccurate analyses and legal complications, underscoring the need for stringent quality standards. The study provides insights into various methods for assessing data quality in GIS, including visual and attribute inspection, spatial analysis, metadata examination, quality control measures, comparative analysis, data validation, and user feedback.

Assessment Methods in GIS [24]: Medeiros complements the discussion by outlining specific assessment methods for data quality in GIS, such as accuracy, completeness, consistency, timeliness, and logical assessments. These methods, whether manual or automated, are crucial for ensuring that GIS data meets the high-quality standards necessary for its intended applications.

In summary, these studies collectively emphasize the multifaceted nature of data quality in GIS, highlighting the importance of comprehensive assessment methodologies to ensure the accuracy, completeness, consistency, and timeliness of geospatial data. The literature underscores the evolving concern for data quality within the GIS community, driven by the increasing reliance on geospatial data for critical decision-making processes. The incorporation of rigorous data quality standards, particularly in participatory GIS projects, is essential for enhancing the reliability and effectiveness of GIS applications in various domains.

3 Methodology

The methodology adopted for evaluating the data quality of Redal GIS involved a systematic approach comprising several key steps demonstrated in Fig. 1.

3.1 Research

Extensive research on data quality assessment principles, both in general and within the context of Geographic Information Systems (GIS). This involved reviewing relevant literature, standards, and best practices in the field of data quality management.

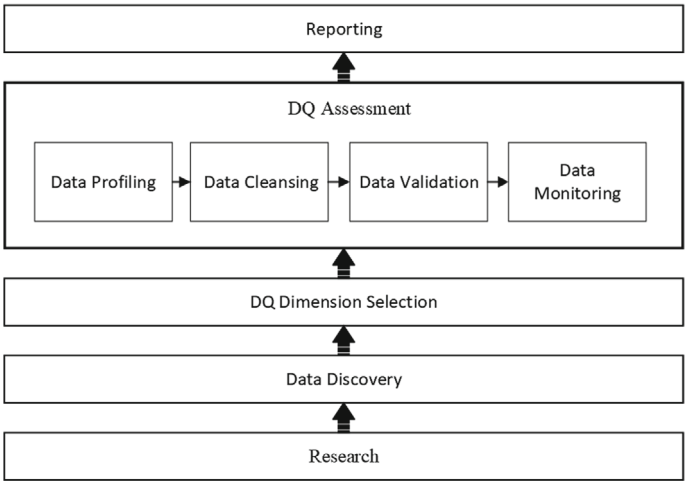


Fig. 1 Pipeline of the adopted DQ assessment methodology

3.2 Data Discovery

Exploring the Redal GIS system to gain a comprehensive understanding of its structure, architecture, and underlying data sources. This step included accessing the GIS platform, examining data schemas, and identifying the sources feeding into the system.

3.3 Dimension Selection

Identification and selection of the relevant data quality dimensions that are crucial for assessing the GIS data effectively. These dimensions encompassed aspects such as accuracy, completeness, consistency, timeliness, and relevancy, tailored to the specific requirements and objectives of Redal's infrastructure management.

3.4 Data Quality Assessment

Carrying out the data quality assessment process, encompassing the following substeps:

1. **Data Profiling**: Examining the structure, content, and quality of the GIS data to gain insights into its characteristics and potential anomalies. This involved analyz-

ing data distribution, identifying outliers, and detecting patterns or irregularities within the dataset.
2. **Data Cleansing**: Addressing inconsistencies, errors, and missing values within the GIS dataset to enhance its overall quality and reliability. This step involved employing data cleaning techniques such as standardization, normalization, deduplication, and error correction to rectify discrepancies and improve data integrity.
3. **Data Validation**: Conducting comprehensive validation checks to assess the accuracy, completeness, consistency, and currency of the GIS data. This entailed comparing the GIS data against authoritative sources, conducting field validations, and performing data validation rules to identify and rectify discrepancies or inaccuracies.
4. **Data Monitoring**: Implementing processes and mechanisms to continuously monitor and maintain data quality over time. This involved establishing data quality metrics, setting up automated validation routines, and instituting data governance practices to ensure ongoing data integrity and reliability within the Redal GIS.

3.5 Reporting

Preparing a comprehensive report summarizing the data quality assessment results, highlighting areas of improvement, and providing actionable recommendations for enhancing data quality within the Redal GIS. The report aimed to facilitate informed decision-making and strategic planning by stakeholders involved in Redal's infrastructure management and operational activities.

4 Data Discovery

Redal GIS stores geographic data about the installed infrastructure in the water distribution, sanitation, and electricity networks in a single Oracle database named SIGREDAL. Each table in SIGREDAL represents a specific asset and it starts with a prefix that indicates which network it belongs to. There are three: ASS for sanitation, AEP for potable water, and ELEC for electricity. The Table 1 indicates the names of all GIS tables that serve as input for this project. (the prefixes for ASS, ELEC and AEP table names were removed for brevity).
The total number of rows for each extracted sample from the tables used as input for this case study amounted to 522,050 entries in total.

Another data source for the project was the WOs (Work Orders) table in Dynamics NAV. This system stores and manages financial and technical data related to Redal's infrastructure and other activities. The WOs table specifically summarizes the financial aspects of projects, making it a valuable resource for identifying projects that have been recorded in the accounting system but not yet in the GIS.

Table 1 GIS Tables

Network	Table name	Definition
AEP	TRONCON	Specific parts of the water distribution infrastructure
	VANNE	Valves, which control the flow and distribution of water
	EQUIPEMENTPUBLIC	Public equipment in the water distribution network
	PIECESPECIALE	Special components within the water distribution network
	CONDUITE	Segments or sections within a sanitation network
	REGARD	Inspection manholes
	REGARDAVALOIR	Gully manholes
	REGARDBORNE	Blind manhole
ASS	REGARDDECHUTE	Drop manholes
	REGARDFACADE	Front manholes
	REGARDGRILLES	Grate manholes
	STATIONPOMPAGE	Pumping stations
	CABLE	Electrical cables
ELEC	COFFRET DIST	Power boxes
	POSTE CLIENT	Customer electrical substations
	POSTE DIST	Distribution substations
	SUPPORT	Support structures

5 Findings

After exploring the Redal GIS, It was concluded that its quality can be assessed through the following dimensions:

5.1 Accuracy

Spatial Accuracy

Spatial accuracy is already guaranteed before inputting any spatial object into the database. This involves comparing the object's coordinates with reference points on the base map to ensure alignment with known geographical features and landmarks. Additionally, the GIS's existing network serves as a crucial reference for assessing spatial accuracy. By cross-referencing object coordinates with the established network infrastructure, professionals can identify any discrepancies or inconsistencies that may affect data integrity.

Table 2 Accuracy assessment examples

Archive code (AC)	Accurate
t31234	Yes
s36789	Yes
b3456	No
r323	No

Table 3 Completeness assessment examples

WO	AC	Installation date	Commissioning date	Length	Type	Complete
12345	t31234	2021-01-15	2021-02-01	100 m	PVC	Yes
23456	s36789	2020-12-01	2021-01-05	80 m	HDPE	Yes
34567	b3456	2021-03-20	–	50 m	–	No

Thematic Accuracy

Thematic accuracy can be assessed by comparing the attribute information stored in the database with its corresponding real-world value. This involves verifying whether the stored value meets the criteria necessary to be considered correct. The attribute information that thematically describes the object is inputted and updated in the database by agents manually copying it from the received scanned descriptive file. Consequently, errors in data accuracy may arise, particularly for properties requiring manual typing. For instance, the archive code (AC) plays a crucial role as the identifier for geographic information about the project. It must adhere to specific criteria; for example, in a sanitation project, the AC should start with either 't3', 'r3', 's3', or 'b3' as a prefix, followed by at least five consecutive digits (0–9), and optionally ending with 'bis' (Table 2).

5.2 Completeness

Completeness can be assessed by ensuring that all essential information about the geographic object is included in the database. Different properties describe each object; for instance, to consider the information about a pipe complete, it should include its work order identifier (WO), archive code (AC), installation date, commissioning date, length, and type (Table 3).

5.3 Consistency

Data consistency can be evaluated by verifying the semantic coherence of the existing values. However, there are currently no constraints in place during the inclusion process to verify data consistency, potentially leading to data inconsistencies. Some proposed consistency verification checks include:

- Commissioning date must be later than the installation date.
- The Work Order (WO) of the object must be present in the work order table of the accounting database.
- There are designated backbone entities such as pipes for sanitation and potable water networks, and cables for the electricity network. Therefore, all work order (WO) and archive code (AC) identifiers of other entities must be included in the backbone beforehand.

5.4 Currency

Currency can be assessed by confirming if the information existing in the database is up-to-date with real-world changes. This can be accomplished by reconciling the Work Order (WO) identifiers of the GIS backbone entities with those in the accounting system. Since every project initiated by Redal is documented in the accounting system database, projects suitable for inclusion in the GIS database must be integrated after completion of the works.

In addition, ensuring currency relies heavily on the accuracy of WO identifiers. An inaccurately spelled WO identifier could mistakenly flag a project as not integrated into GIS when it actually is. This underscores the critical importance of ensuring the accuracy of WO identifiers.

Data Quality Metrics

For completeness, accuracy, consistency, and currency, metrics were calculated as follows:

Completeness:
For a DataFrame DF with key attributes $A_1, A_2, \ldots, A_n$ where n is the number of key attributes, the completeness score COM is calculated as follows:

$$COM(DF) = \frac{\sum_{i=1}^{n} COM(DF(A_i))}{n} \tag{1}$$

where $COM(DF(A_i))$ is the completeness score of the column A_i of the DataFrame DF, calculated as follows:

$$COM(DF(A_i)) = \frac{len\ DF(A_i)_{non-null}}{len\ DF(A_i)} \tag{2}$$

where $len\ DF(A_i)_{non-null}$ is the number of non-null values in the column A_i of the DataFrame DF

Accuracy:
For a DataFrame DF with key Free-text attributes $A_1, A_2, \ldots, A_n$ where n is the number of key attributes, the accuracy score ACC is calculated as follows:

$$CON(DF) = \frac{\sum_{i=1}^{n} ACC(DF(A_i))}{n} \tag{3}$$

where $ACC(DF(A_i))$ is the accuracy score of the column A_i of the DataFrame DF, calculated as follows:

$$ACC(DF) = \frac{len\ DF(A_i)_{accurate}}{len\ DF(A_i)} \tag{4}$$

where $len\ DF(A_i)_{accurate}$ is the number of accurate (non null) values in the column A_i of the DataFrame DF

Consistency:
For a DataFrame DF with attributes $A_1, A_2, \ldots, A_n$ where n is the number of key attributes, the consistency score $CONS$ is calculated as follows:

$$CONS(DF) = \frac{\sum_{i=1}^{n} CONS(DF(A_i))}{n} \tag{5}$$

where $CONS(DF(A_i))$ is the consistency score of the column A_i of the DataFrame DF, calculated as follows:

$$CONS(DF) = \frac{len\ DF(A_i)_{consistent}}{len\ DF(A_i)} \tag{6}$$

where $len\ DF(A_i)_{consistent}$ is the number of consistent (non null) values in the column A_i of the DataFrame DF

Currency:
The currency score for the DataFrame DF, calculated as follows:

$$CUR(DF) = \frac{\sum_{j=1}^{m} (t_j - t_j^{blocked})}{m}$$

Here, m represents the total number of objects in the DataFrame DF, t_j represents the time at which data for object j were stored in the GIS, and $t_j^{blocked}$ represents the time at which data for object j were marked as blocked in the accounting system.

Table 4 DQ scores across potable water network assets with averages

Asset	Currency	Accuracy	Completeness	Consistency	Global Score
Public equipment	70.88	61.81	44.65	98.12	68.87
Special component	85.24	67.07	46.49	100.00	74.7
Pipe	62.44	61.16	66.12	88.56	69.57
Valve	66.12	61.37	48.12	92.11	66.93
Global score	71.17	62.85	51.35	94.70	70.02

The Table 4 provides a sample for the global calculated scores across some of the tables corresponding to the potable network.
Note None of the metrics below represent the actual ones.

These findings underscore the necessity of adopting robust data quality frameworks and the potential benefits of system integrations to enhance the overall utility of GIS within Redal.

6 Discussion

Enhancing the quality of attribute data within Redal GIS involves a multifaceted approach. Firstly, identifying outdated or erroneous data points is essential. This could include conducting regular audits or data validation checks to flag inconsistencies or inaccuracies. Once identified, these data points should be promptly updated or corrected to ensure accuracy.

Regular audits or data validation checks can serve as effective mechanisms for identifying outdated or erroneous data points within the Redal GIS. By systematically reviewing the attribute data, inconsistencies or inaccuracies can be flagged and addressed in a timely manner. This proactive approach helps maintain the accuracy and reliability of the GIS database, ensuring that decision-making processes based on this data are well-informed.

Additionally, implementing robust constraints and rules for the inclusion of new data is crucial. This involves establishing clear guidelines and standards for data entry, ensuring consistency and coherence across all attributes. For example, defining specific formatting requirements or validation rules for data entry fields ensuring that only accurate and relevant data is incorporated into the database, further enhancing its quality and usability.

Thus, enhancing the quality of attribute data within the Redal GIS requires a comprehensive approach that includes regular audits, data validation checks, and the implementation of robust constraints and rules for data inclusion. By adopting these measures, Redal can ensure the accuracy, reliability, and integrity of its GIS database, ultimately supporting informed decision-making and effective planning processes.

7 Conclusion

In conclusion, this paper has explored the assessment of data quality within the Redal GIS, focusing on dimensions such as accuracy, completeness, consistency, and currency. The findings revealed that while the Redal GIS serves as a critical tool for decision-making and effective planning, there are areas for improvement in data quality. The assessment of accuracy highlighted the importance of spatial and thematic accuracy, emphasizing the need for rigorous verification processes to ensure alignment with real-world data. Completeness assessment underscored the necessity of including all essential information about geographic objects to enhance the usability of the GIS database. Moreover, the evaluation of consistency revealed the importance of implementing constraints and rules to maintain data coherence and prevent errors. Currency assessment emphasized the need for regular updates and reconciliation with external databases to ensure the timeliness of the information. To address these challenges and enhance data quality within the Redal GIS, recommendations include conducting regular audits, implementing validation checks, and establishing robust constraints and rules for data inclusion. By adopting these measures, Redal can improve the reliability and integrity of its GIS database, ultimately supporting informed decision-making and effective infrastructure management. Moving forward, continuous monitoring and evaluation of data quality will be essential to ensure the ongoing effectiveness of the Redal GIS. Additionally, fostering a culture of data stewardship and providing training for personnel involved in data management will further contribute to maintaining high standards of data quality. Overall, this study contributes to the literature on GIS data quality assessment and provides practical insights for utility companies like Redal to enhance the reliability and usability of their GIS databases. By prioritizing data quality, organizations can optimize their decision-making processes and better serve their stakeholders and communities. This work paves the way for potential improvements, using techniques published in [1–3, 6, 10, 12–18, 20, 31, 32, 36, 37], which will make it possible to meet these challenges and further optimize the performance of the System.

References

1. Abourezq, M., Idrissi, A.: A Cloud Services Research and Selection System. IEEE ICMCS (2014)
2. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Arch. **9**(2–3), 136–148 (2020)
3. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless ad hoc networks using the skyline operator and an outranking method. In: Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
4. Ahmed, M.M., Soo, W.L., Hanafiah, M.A.M., Ghani, M.R.A.: Development of customized distribution automation system (das) for secure fault isolation in low voltage distribution system. In: Guedes, L.A. (ed.) Programmable Logic Controller, chapter 8. IntechOpen, Rijeka (2010)

5. Batini, C., Cappiello, C., Francalanci, C., Maurino, A.: Methodologies for data quality assessment and improvement. ACM Comput. Surv. **41**(3) (2009)
6. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on html tags. J. Theor. Appl. Inform. Technol. **55**(1), 137–148 (2013)
7. Bouhadjar, M.: Quality Assessment of Geospatial Data (2014)
8. SMQ-RSE Department: Rapport Développement Durable: Média5. Redal **11**, 2022 (2021)
9. Devillers, R., Bédard, Y., Jeansoulin, R.: Multidimensional management of geospatial data quality information for its dynamic use within GIS. Photogram. Eng. Remote Sens. **71**(2), 205–215 (2005)
10. Elhandri, K., Idrissi, A.: Parallelization of top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2021)
11. Fenais, A., Ariaratnam, S., Ayer, S., Smilovsky, N.: Integrating geographic information systems and augmented reality for mapping underground utilities. Infrastructures **4**, 60 (2019)
12. El handri, K., Idrissi, A.: Comparative study of topk based on Fagin's algorithm using correlation metrics in cloud computing GOS. Int. J. Internet Technol. Secur. Trans. **10** (2020)
13. El Handri, K., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
14. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv preprint arXiv:1307.5910
15. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inform. Technol. **37**(2), 141–158 (2012)
16. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. In: International Conference on Big Data and Advanced Wireless Technologies (2016)
17. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF 107–116 (2006)
18. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
19. Imane, Z., Hachmi, M.K.B., Halbac-Cotoara-Zamfir, R.: Quantitative water management in Rabat, sale and Timisoara drinking water system. Environ. Eng. Manage. J. **18**(12), 2567–2577 (2019)
20. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
21. Luu, H.: Spark SQL (foundations). In: Beginning Apache Spark 2, pp. 87–145. Apress (2018)
22. Manley, E.D.: Dash: an easy-to-use framework for building web applications and dashboards. J. Comput. Sci. Coll. **38**(6), 104–105 (2023)
23. McCluskey, J.: Philosophy and design of reverse osmosis membrane replacement (2020)
24. Medeiros, G., Holanda, M.:. Solutions for data quality in GIS and VGI: a systematic literature review. In: Rocha, A., Adeli, H., Reis, L.P., Costanzo, S. (eds.) New Knowledge in Information Systems and Technologies, pp. 645–654. Springer International Publishing, Cham (2019)
25. Meyers, J.: Gis in the utilities. In: Longley, P.A., Goodchild, M.F., Maguire, D.J., Rhind, D.W. (eds.) Geographical Information Systems: Principles, Techniques, Management and Applications, Chapter 57, 2nd edn, pp. 801–818. Wiley, Abridgedition edition (2005)
26. Mishra, R.K.: PySpark Recipes. Apress (2018)
27. Mukhopadhyay, S.: Introduction. In: Advanced Data Analytics Using Python, pp. 1–22. Apress (2018)
28. Musungu, K.: Assessing spatial data quality of participatory GIS studies: a case study in cape town. ISPRS Ann. Photogrammetry Remote Sens. Spatial Inform. Sci. II-2/W2, 75–82 (2015)

29. Nie, S.-J., Xiong, P.-F., Qing, C.-Q., Huang, H.-F.: Different models analysis of project lifecycle. In: 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing, pp. 1–4 (2008)
30. Radut, C., Badescu, A.: Geographic information systems and business environments. Revista Economia Contemporană 2(4) (2017)
31. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Mining (2017)
32. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular ad-hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
33. Rolon-Mérette, D., Ross, M., Rolon-Mérette, T., Church, K.: Introduction to anaconda and python: installation and setup. Quant. Methods Psychol. **16**(5), S3–S11 (2020)
34. Veregin, H.: Data quality parameters. In: Longley, P.A., Goodchild, M.F., Maguire, D.J., Rhind, D.W. (eds.) Geographical Information Systems: Principles, Techniques, Management and Applications, chapter 12, 2nd edition, pp. 117–189. Wiley, abridgedition edition (2005)
35. Widad, E., Saida, E., Gahi, Y.: Quality anomaly detection using predictive techniques: an extensive big data quality framework for reliable data analysis. IEEE Access **11**, 103306–103318 (2023)
36. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in moocs using topic modeling and nlp techniques. Int. J. Educ. Inform. Technol. 5567–5584 (2023)
37. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mob. Robot. Intell. Syst. **14**(3), 65–70 (2020)

ELK Stack Approach with Artificial Intelligence for Logs Collection and Resource Usage Monitoring and Forecasting

Khawla Elansari, Abdellah Idrissi, and Kaoutar Moutaouakil

Abstract In today's fast-paced digital world, businesses constantly seek innovative solutions to monitor their systems, gather logs, and adequately manage resource utilization. These responsibilities are critical for guaranteeing the smooth functioning of numerous apps and services by assuring a seamless and optimal operation. This paper presents an end-to-end approach to effectively introduce an artificial intelligence-based resource usage forecasting method into an enterprise environment. This solution begins with collecting and storing metrics data using the ELK stack and progresses to forecasting utilizing the power of Artificial Intelligence, notably Deep Learning. The Elasticsearch, Logstash, and Kibana (ELK) stack is widely considered a complete and valuable platform for log gathering, processing, and visualization. It becomes much more potent when integrated with AI in estimating resource utilization and anticipating future trends.

1 Introduction

Organizations face a pressing challenge in an era marked by the persistent expansion of data and the resulting need for solid information technology (IT) infrastructure: the effective collection, centralization, and analysis of system logs and metrics. This paper provides a thorough examination of a multidimensional solution customized to this requirement based on the convergence of proven industrial practices and cutting-edge artificial intelligence (AI) approaches.

Implementing the ELK Stack, a composite architecture comprised of Elasticsearch, Logstash, and Kibana, which have emerged as essential core components of modern log and metric management systems individually and collectively, is critical to this attempt. This powerful trio combines the imperatives of data gathering, transformation, and visualization.

K. Elansari (✉) · A. Idrissi · K. Moutaouakil
Computer Science Laboratory (LRI), Computer Science Department, Faculty of Science of Rabat, IPSS Team, Mohammed V University in Rabat, Rabat, Morocco
e-mail: khawla.elansari@um5r.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_41

This article also discusses deep learning, a burgeoning field within AI, and its implications for predictive resource usage and detecting imminent system problems. The convergence of machine learning and deep learning approaches adds predictive modeling skills to the repertory of IT infrastructure management, proactively preventing service interruptions and optimizing resource allocation.

This study article navigates the complex realm of centralized system log and metric administration, illuminating the symbiotic relationship between traditional technologies and emergent AI paradigms. We investigate this integrated approach's theoretical foundations, methodological complexities, and empirical consequences. This synthesis increases our understanding of IT infrastructure management and presents a unique technique that can potentially change resource optimization and incident mitigation practices.

2 Related Work

Much research has been conducted on pivotal issues of centralized system log and metric management and predicted resource utilization in IT infrastructure. Previous research has examined the various components of the ELK Stack, including Elasticsearch for log indexing and search capabilities [1], Logstash for log ingestion and processing [2], and Kibana for real-time data visualization [3].

These investigations have provided a thorough grasp of the strengths and limitations of these components. Furthermore, much research has been conducted on machine learning applications in resource usage predictions. Among these is the work that proved the usefulness of regression-based machine learning models in forecasting server resource utilization with exceptional accuracy [4].

In this regard, another research broadened the realm of predictive capabilities by including deep learning techniques, resulting in a recurrent neural network (RNN) architecture that demonstrated excellent performance in resource forecasting tests [5].

At the same time, important work highlighted the relevance of real-time anomaly detection in assuring system dependability and developed an unsupervised learning framework for anomaly identification in log data [6]. So far, academic discourse has supplied a significant basis for this study to unite these dissimilar spheres synergistically.

2.1 Cloud Monitoring Systems

Cloud monitoring examines, regulates, and manages the operational workflow and processes inside a cloud architecture. It uses manual or automated IT monitoring and management strategies to ensure the best performance of a cloud infrastructure or platform. It is a critical tool for regulating and managing cloud infrastructure

Fig. 1 CMS Phases

by collecting data from various probes, aggregating related data, filtering unrelated/unwanted data, and analyzing or evaluating cloud performance. It also conducts control activities to improve cloud performance.

Cloud Monitoring Systems aid in cloud performance management, mainly when consumer-adapted vital services or scientific applications are used. For example, if a consumer wants to run an application on different clouds to ensure high availability, CMS simplifies moving between clouds. Cloud computing platforms require specific CMS for practical usage, administration of their enormous complexity, and guarantee of suitable Quality of Service (QoS) levels. Furthermore, CMS provides information to consumers and providers such as workload, QoS parameters, key performance indicators, and resource use status. This contributes to billing transparency between provider and consumer.

CMSs are classified into two sorts based on their architecture: centralized and decentralized. A single monitoring server collects measurements from multiple nodes and saves the data in centralized storage for further processing in the centralized architecture. The centralized design is inexpensive but has two drawbacks: one point of failure and a need for more scalability.

A decentralized design is implemented to circumvent these issues, with monitoring tasks dispersed among several cloud nodes. The monitoring architecture may be created using either an agent-based or an agentless system.

The monitoring activity of CMS is divided into five phases (Fig. 1).

2.2 ELK Stack

The ELK Stack (Elasticsearch, Logstash, and Kibana) is a collection of open-source applications that search, analyze, and display data from any source and format in real-time. It provides a next-generation log management platform that overcomes log heterogeneity and scale concerns.

2.2.1 Elasticsearch

Elasticsearch is an Apache Lucene-based search engine. It is a real-time, distributed, multitenant- capable full-text search engine. It offers a RESTful API based on JSON documents. It can be used for full-text search, structured search, analytics, or all three. One of its most important advantages is the capacity to search quickly by indexing the text to be searched. Many search engines have long been available with

the option to search by timestamp or precise quantities, Elasticsearch distinguishes itself by running full-text searches, managing synonyms, and evaluating items based on relevancy.

2.2.2 Logtash

Logstash is the next-generation logging framework; it functions as a centralized framework for log collecting, processing, storage, and search; it can normalize data from several sources and dynamically combine them into your chosen destinations.

With a wide range of input, filter, and output plugins, Logstash enables any event to be enhanced and altered with many native codecs that simplify the ingestion process. By utilizing more data, both in terms of volume and diversity, Logstash offers insights.

Files, Syslog, TCP/UDP, stdin, and many other input methods are just a few of the input sources that Logstash may accept. A wide variety of filters may be used to change the events in the gathered logs [2].

2.2.3 Kibana

A data visualization platform primarily used to analyze massive volumes of logs in line graphs, bar graphs, pie charts, heat maps, region maps, coordinate maps, gauges, goals, and other visual representations. Thanks to the display, it is simple to foresee or notice changes in trends of mistakes or other noteworthy events of the input source [3].

3 Methodology

3.1 Data Collection and Ingestion

We aim to gather information from many sources and consolidate it in the ELK stack for analysis to get a complete picture of the IT environment. This aids in finding performance bottlenecks, possible security issues, and resource allocation optimization.

The company's infrastructure is distributed across several servers, each of which plays a vital part in how our clients operate. To guarantee availability, optimize resource allocation, and proactively handle possible issues, these servers must be monitored along with the data points they generate.

3.1.1 Azure SQL Database

The data collection's main objective is to obtain critical infrastructure-related data kept in Azure SQL Database. The server specifics, settings, performance metrics, and other pertinent data are all included in this report and are used to support our entire IT monitoring strategy.

Here is the process to get this data:

- Logstash is installed on a virtual machine (VM) in Azure; the Azure SQL Database is queried for server information via SQL queries that get details including server names, setups, hardware specs, system CPU and RAM metrics, and any other data elements essential to our infrastructure monitoring utilizing the JDBC input plugin. The data is subsequently processed using Logstash.
- The data that Logstash has processed is sent to the ELK Stack and stored in Elasticsearch, then analyzed in Kibana using a user-friendly interface.

3.1.2 Metrics from Metricbeat Agents

Collecting system metrics and performance data from various servers and devices inside the IT environment on-premise is the focus of our second data collection goal. Metricbeat Agents are strategically placed to record critical metrics, including CPU utilization, memory consumption, disk I/O, and network activity.

Proactively tracking system metrics is essential to the smooth running of our IT infrastructure. We want to identify abnormalities, anticipate prospective problems, and improve resource allocation by gathering these indicators.

Metricbeat agents set up on all pertinent servers and devices continually gather system metrics and performance data. These agents are set up to send the data gathered to our ELK stack.

Here is the process to get these metrics:

- Two Logstash instances serve as the data processors and data collectors for the on-premises data. These Logstash instances take in the data that Metricbeat has gathered.
- The Logstash instances are distributed with incoming data from on-premises servers using a load balancing layer (HAProxy). In addition to ensuring high availability and fault tolerance, this helps divide the burden.
- Data from the load balancer is transmitted to a Logstash instance operating within an AWS container. The containerized instance of Logstash serves as a middleman for the processing and transforming of data.
- The containerized Logstash instance sends the data to the ELK Stack on AWS for archiving, processing, and visualization.

3.2 Data Analysis

3.2.1 Elasticsearch as Our Leading Data Store

Elasticsearch data analysis fundamentals include:

- Indexing and mapping data for enhanced search efficiency.
- Utilizing full-text search for efficient problem identification in logs and events.
- Leveraging advanced aggregation for comprehensive data analysis, enabling detection of trends, patterns, and anomalies. Aggregations facilitate not just item identification but also detailed analyses like item counts, median and average calculations, and multi-dimensional analyses (e.g., by manufacturer). They also aid in tracking historical trends, identifying top manufacturers, and discovering anomalies that may reveal underlying patterns or issues.

3.2.2 Visualization of Data Using Kibana

For visualization and exploration purposes, Kibana offers features including:

- Creation of custom dashboards to display real-time information like server performance metrics, log trends, and security alerts.
- Ad-hoc querying capabilities through Kibana's intuitive query language and filters allow for spontaneous data investigation and anomaly analysis.
- Reporting and alerting mechanisms in Kibana notify IT teams about critical incidents or security issues.

3.2.3 Real-Time Logs Analysis

Real-time log analysis is necessary for tracking server health and swiftly detecting problems:

- Before logs are indexed in Elasticsearch, Logstash examines them from various sources to ensure consistency and formatting.
- Logstash transforms unstructured logs into structured data that can be used for analysis and querying.
- We can build dashboards centered on logs using Kibana's visualization features, which provide insights into application behavior, error patterns, and performance bottlenecks.

3.3 Data Monitoring and Alerting

3.3.1 Infrastructure Monitoring in Real Time

The first line of protection against possible interruptions is effective monitoring. Real-time monitoring's main components are:

- Server health metrics are continuously collected, including CPU, RAM, disk I/O, and network traffic.
- Monitoring application performance indicators helps to provide the best functionality and responsiveness.
- Monitoring resource use will help us spot any possible bottlenecks and allocate resources more effectively.

3.3.2 Event and Log Monitoring

Comprehensive log and event monitoring sheds light on system behavior and spots problems early on.

- Monitoring logs in real-time for security incidents, abnormalities, and error patterns.
- Correlating log events to find possible systemic problems and fundamental causes.
- Log analysis is used to find suspicious activity, illegal access, and security breaches.

3.3.3 Escalation and Management of Incidents

Timely and efficient responses are ensured through an established incident escalation and management process:

- Classifying and ranking occurrences according to their gravity and significance.
- Automation of predetermined incident reactions, such as resource scaling or service restarts.
- Alerts are delivered via various notification methods, including email and interaction with teamwork applications like Slack or Microsoft Teams.
- Monitoring the status of incident resolution and identifying recurrent problems through incident tracking and reporting.

3.3.4 Customizable Dashboards

Kibana's customizable dashboards offer a consolidated view of monitoring data:

- IT teams may design custom dashboards that are suited to their monitoring requirements.

– Dashboards are updated in real-time to reflect the most recent data, ensuring that insights are provided on time.
– Data visualization: graphs that make measurements, logs, and alarms easier to read.

3.3.5 Alerting

Alerting is done in real-time using "Watcher" or "Alerting rules", a component of the ELK stack. It continually evaluates incoming data and sends alarms when certain conditions are met.

IT personnel can be informed via alerts about abnormalities, security breaches, or performance problems.

4 Implementation

4.1 Centralized Log and Metric Collection with ELK Stack

The creation of a centralized system for gathering and analyzing system logs and metrics was accomplished using the ELK Stack—Elasticsearch, Logstash, and Kibana, as illustrated in Fig. 2.

Elasticsearch functioned as the primary data repository, capable of ingesting and indexing the massive volumes of log and metric data created by our IT infrastructure. Its real-time search capabilities assured that data could be accessed and examined quickly, while its scalability ensured that we could handle our ever-increasing data quantities.

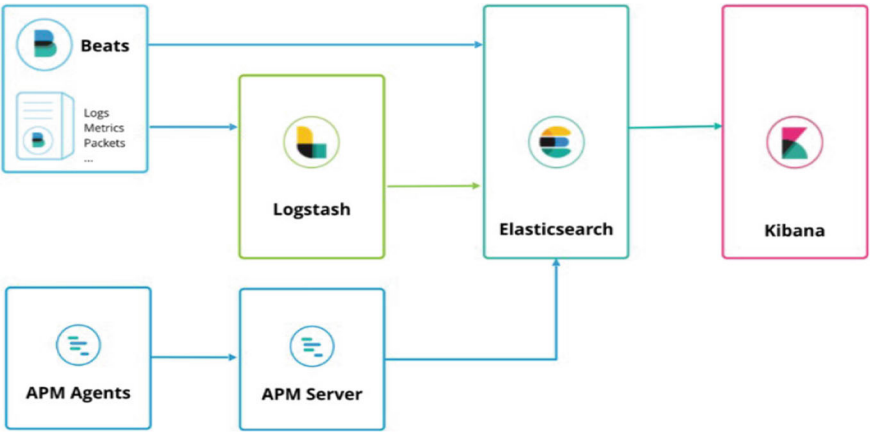


Fig. 2 ELK stack

Logstash, in collaboration with Elasticsearch, performed the critical task of data intake. It processed, altered, and enhanced incoming data from various sources, allowing for data standards and structural consistency.

The capacity of Logstash to perform complicated data transformations was critical in preparing our data for meaningful analysis (Fig. 3).

The Kibana component also provided a visually appealing data exploration, analysis, and visualization interface. Our IT teams were able to receive real-time insights into the system performance, track problems, and develop graphical representations of crucial data using Kibana (Fig. 4).

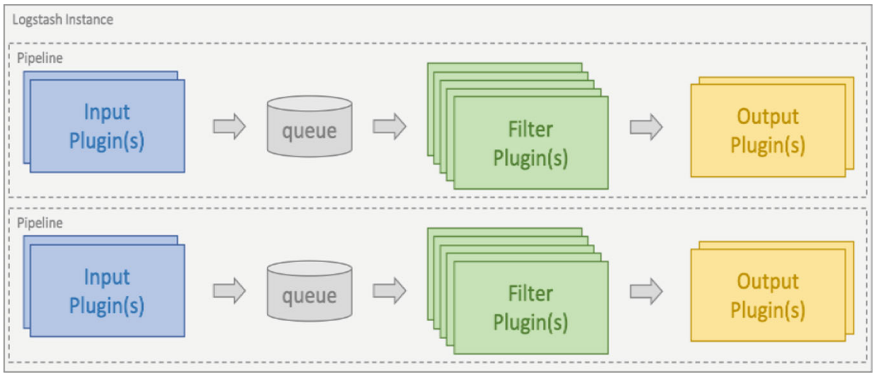


Fig. 3 Logstash Instance

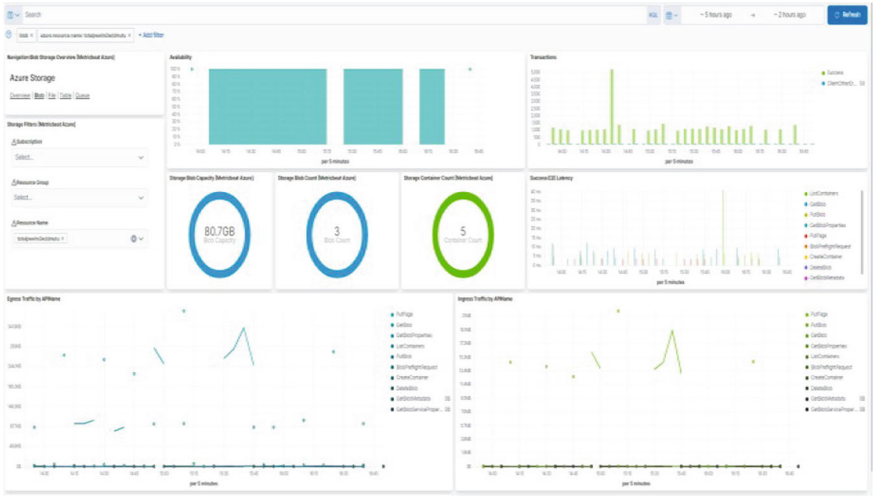


Fig. 4 Kibana dashboard

The combination of Elasticsearch, Logstash, and Kibana resulted in an integrated solution that not only consolidated our log and metric data but also gave us the toolset we needed to navigate the complexities of our IT architecture easily.

4.2 Machine Learning and Deep Learning Integration for Resource Forecasting

In our pursuit of predictive resource usage, we investigated numerous machine learning and deep learning models. This round of testing was critical in selecting the best method for our project. Autoregressive Integrated Moving Average (ARIMA), XGBoost, Long Short-Term Memory Networks (LSTM), and Convolutional Neural Networks (CNN) were among the models tested. Each model was rigorously trained and validated using historical resource consumption data to test predicting accuracy [7] (Table 1).

We compared the machine learning and deep learning models implemented on the Mean Absolute Error (MAE) indices, which measure mistakes between paired observations describing the same phenomena.

With an MAE of just 5.03, Autoregressive Integrated Moving Average (ARIMA) has shown excellent accuracy. This indicates that ARIMA effectively captures temporal relationships and seasonality in resource consumption data, making it a solid option for this task.

With a relatively low MAE of 870.13, XGBoost worked admirably. It has demonstrated its capacity to model complicated relationships within data. While not as exact as ARIMA or LSTM, it provides a decent combination of complexity and performance.

LSTM outshines all other models with an extraordinarily low MAE of 0.825. This remarkably accurate forecasting indicates that LSTM effectively captures the intricate patterns in resource utilization data, excelling in handling the dynamic and sequential nature of the data.

On the other hand, CNN's extremely high MAE rating shows that it is unsuitable for this task. The astronomical inaccuracy shows a significant mismatch between

Table 1 Models' evaluation

Model	Measure		
	MAE Mean absolute error	MAP Mean squared error	RMSE Root Mean squared error
ARIMA	5.02	325.44	18.04
XGBoost	870.13	4,563,967.97	2136.34
LSTM	0.82	7.53	2.74
CNN	1,069,651,160,309.97	8.73e + 24	2,956,316,605,941.47

CNN's design and the resource consumption statistics, rendering it unfit for this application.

4.3 Discussion

This study aims to alter the world of IT infrastructure management through the strategic combination of centralized log and metric management enabled by the ELK Stack and the predictive capabilities of machine learning and deep learning. Several essential aspects were investigated to improve resource consumption forecasts and yield deep insights that have the potential to alter how corporations monitor and manage their IT ecosystems.

The ELK Stack implementation—Elasticsearch, Logstash, and Kibana—is the foundation of this revolutionary journey. The strength of Elasticsearch's data indexing, Logstash's data ingestion, and Kibana's data visualization are combined to produce an integrated system capable of smoothly gathering and analyzing disparate data sources. This fundamental design enables IT professionals to respond to crises proactively and improve resource allocation, enhancing overall operational efficiency and resilience of IT infrastructure.

A comparison of machine and deep learning models in resource usage forecasts reveals striking insights. While classic approaches such as ARIMA produce acceptable results, more modern techniques outperform them. Because of its capacity to capture complicated correlations in data, XGBoost, a gradient-boosting algorithm, emerges as a viable competitor. Nonetheless, LSTM, a recurrent neural network, outshines them with an extremely low Mean Absolute Error (MAE) of 0.825. This accomplishment demonstrates LSTM's unrivaled ability to comprehend the complicated patterns of resource usage data.

These efforts result in an integrated method combining the ELK Stack's centralized log and metric management capabilities with LSTM's resource forecasting skills. The importance of this integration cannot be overstated. It enables businesses to anticipate concerns and events, lowering the risk of resource depletion and increasing operational efficiency. IT infrastructure management is elevated to a more strategic and agile level by employing modern artificial intelligence and machine learning approaches. This transformation is vital in an era when enterprises' digital operations have never been more critical.

The integration of centralized log and metric management with LSTM-driven resource usage forecasts is a critical outcome of this trip. This convergence marks a fundamental change in the management of IT infrastructure. Proactive issue detection and prevention, along with optimal resource allocation, results in significant increases in operational efficiency and the resilience of IT systems.

The strategic integration of the ELK Stack with sophisticated AI approaches transforms IT infrastructure management from reactive to proactive, matching it with the demands of an increasingly digitalized environment.

Looking ahead, we see this transformational journey as a dynamic progression rather than a static endpoint. There are several opportunities for further investigation, including incorporating new machine learning and deep learning models, creating hybrid techniques, and expanding the integrated approach into other aspects of IT infrastructure management. In addition we think that we can use approaches proposed in [8–24] to improve the system.

In conclusion, this study provides a framework for enterprises to adopt a forward-thinking approach to IT infrastructure management, promoting dependability, resource optimization, and proactive risk reduction. The digital world is constantly changing, and companies are well-equipped to tackle the challenges and possibilities with the tools and insights given here.

References

1. Clinton, G., Zachary, T.: Elasticsearch: The Definitive Guide. O'Reilly Media (2015)
2. https://www.elastic.co/fr/logstash. Logstash Centralize, transform and stash your data
3. https://www.elastic.co/kibana. Kibana: Explore, Visualize, Discover Data
4. Malik, S.,MTahir, M., Sardaraz, M., Alourani, A.: A Resource Utilization Prediction Model for Cloud Data Centers Using Evolutionary Algorithms and Machine Learning Techniques (2022)
5. Alasaly, M.S., Bencherif, M.A., Alsanad, A., Hassan, M.M.: A deep learning-based resource usage prediction model for resource provisioning in an autonomic cloud computing environment (2022)
6. Landauer, M., Onder, S., Skopik, F., Wurzenberger, M.: Deep Learning for Anomaly Detection in Log Data: A Survey (2022)
7. Amiri, M., Mohammad-Khanli, L.: Survey on prediction models of applications for resources provisioning in cloud (2017)
8. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2021) 2020
9. Elhandri, K., Idrissi, A.: Comparative study of Top_k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Internet Technol. Secured Trans. **10** (2020)
10. ElHandri, K., Idrissi, A.: Parallelization of Top_k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
11. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Mining (2017)
12. Idrissi, A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv preprint arXiv:1307.5910
13. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theor. Appl. Inf. Technol. **37**(2), 141–158 (2012)
14. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
15. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M : Top-k and skyline for cloud services research and selection system. International Conference on Big Data and Advanced Wireless Technologies (2016)
16. Idrissi, A, Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF, 107–116 (2006)

17. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mobile Robot. Intell. Syst. **14**(3), 65–70 (2020)
18. Abourezq, M., Idrissi, A.: A Cloud Services Research and Selection System. IEEE ICMCS (2014)
19. Abourezq, M., Idrissi, A., Yakine, F: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
20. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
21. Laghrissi, A., Retal, S., Idrissi, A: Modeling and optimization of the network functions placement using constraint programming. Proceedings of the International Conference on Big Data and Advanced Wireless Technologies (2016)
22. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
23. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theor. Appl. Inf. Technol. **55**(1), 137–148 (2013)
24. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)

Efficient Wireless Communication in Mobile Edge Computing: Channel Allocation Problem

Sara Maftah, Mohamed El Ghmary, and Mohamed Amnai

Abstract Mobile Edge Computing is a promising computing paradigm due to its proximity to the end user and the ability to offload computation to an edge server. The main feature that distinguishes Mobile Edge Computing from Cloud and Fog computing is computation offloading, it enables mobile users to offload their workload to be processed in a fast and efficient way. Nowadays, wireless devices are in a constant evolution, whether in terms of the hardware or the software. However, the applications running on these devices require high computing and storage resources. In this process, data must be uploaded and downloaded, which requires high bandwidth and multiple network channels in order to deploy a channel allocation strategy for optimization purposes. In a previous work, we examined the impact of network delay on offloading tasks from a mobile device to a nearby edge server, this delay is caused by network channels and their bandwidth. In this paper, we study network modeling in a cellular network to improve the quality of wireless communication in Mobile Edge Computing environments.

Keywords Mobile edge computing · Computation offloading · Channel allocation · Internet of things

S. Maftah (✉) · M. Amnai
Faculty of Sciences, Ibn Tofaïl University, Kenitra, Morocco
e-mail: sara.maftah@uit.ac.ma

M. Amnai
e-mail: mohamed.amnai@uit.ac.ma

M. El Ghmary
FSDM, Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: mohamed.elghmary@usmba.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_42

1 Introduction

Driven by the fast exponential evolution of the Internet of Things (IoT), the flaws in the existing paradigms such as Cloud and Fog computing persist and impose great constraints to obtain satisfying results due to the demand for real time processing and execution. IoT devices are known for being less efficient when it comes to processing heavy tasks that require a certain high computation resource, they are also constrained by their battery management system and ability to maintain a longer lifetime. In an attempt to help IoT devices to process the heavy tasks, computation offloading was introduced early for energy saving purposes in which they relied on a Cloud architecture [1], however, due to latency constraint, a new computing paradigm was needed to ensure a faster response time. Therefore, Mobile Edge Computing (MEC) was introduced to cover and offer a better performance [2].

To improve the reliability of low-power IoT, the research community has covered myriad ways of optimization methods for computation offloading in Mobile Edge Computing, whether in terms of minimizing communication or computation costs.The architecture of Mobile Edge Computing relies on deploying edge servers within the Radio Access Network (RAN), which is considered the closest placement to IoT devices. These deployed servers host multiple virtual machines and are managed in a way not only to minimize latency by being in close proximity to IoT devices but also ensures an optimized resource allocation, thereby enhancing the overall performance and responsiveness of the MEC system [3, 4].

Computation offloading is the process of executing a Mobile device's task on a different computing resource, whether on an Edge Server or another Mobile device nearby with enough resources. It is classified into binary (full) and partial offloading [2, 5], the choice between these two modes depends on many factors such as the nature of the computation and tasks, available computing and communication resources and energy constraints [6].

Computation offloading to MEC servers is a crucial mechanism to improve the quality of service and offer a better user experience. Optimization methods for computation offloading imply finding the optimal offloading decision [7], which is challenging due to different constraints [8]. Network delay and latency are important factors when it comes to computation offloading. Hence, the state of a wireless channel is one of the influential constraints that should be taken into consideration. Mobile communication, according to [9], is a vital part to design an efficient overall computing paradigm and it must support the additional requirements in terms of bandwidth and delay, which is done by teaming up with the 5th Generation of cellular networks (5G) that provides mobility, high bandwidth, low latency and low battery consumption [10].

Researchers have proposed communication models to formulate computation offloading system models. According to [11] improving communication reliability for IoT devices can be done by adopting a multichannel communication approach which reduces the effects of channel impairments such as interference. [12] also noticed that interference may be incurred due to network range reuse, which will

significantly deteriorate network performance, hence, in order to improve the performance of wireless cellular networks with Mobile Edge Computing, they proposed an integrated framework to jointly consider computation offloading, resource allocation and content caching. Aiming to optimize system efficiency and deliver improved outcomes for next-generation wireless networks, the authors in [13] studied the challenges posed by channel interference and time slot multi-user collisions that arise from radio communication in wireless networks during information transmission. Channel interference occurs when multiple devices communicate simultaneously on the same frequency, resulting a signal degradation and reduced throughput, in this context, they introduced a novel one-to-one matching algorithm that leverages Pareto improvement to prioritize solutions that benefit at least one device without compromising the performance of others, and swapping operations to allow a dynamic reassignment of channels between devices.

Moreover, the authors in [14] adopted a system designed to optimize both task latency and device energy consumption in multi-user MEC systems. The allocation strategy employed in their system takes advantage of the diversity gain offered by the wireless channel to enable efficient task offloading with reduced transmission delay.

The authors in [15] have also addressed the latency problem by jointly allocating wireless and computational resources to facilitate computation offloading in MEC, their aim consists on minimizing the maximum delay for offloading and computing.

A MEC paradigm does not only aim to provide computing resources for the computation-intensive devices but also places a significant emphasis on reducing latency, which is a critical aspect highlighted by [16] alongside transmission rate as the principal Quality of Service (QoS) indicators. The authors adopted a Network Slicing Model to increase the processing efficiency, thus satisfying the Quality of Service demands and reinforcing the adaptability of the system to the dynamic needs of computation-intensive tasks.

Optimizing task latency is crucial in MEC systems, as it directly impacts the responsiveness and user experience of applications running on the devices. Therefore, in this paper, we aim to delve into the various aspects of network modeling.

After introducing the theme and providing an overview of the state of art, the remainder of the paper is structured as follows. In Sect. 2, we will go through computation offloading modeling briefly to focus on the specifics of channel modeling. The simulation and results are discussed in Sect. 3. Finally, Sect. 4 is the conclusion of this paper.

2 Channel Modeling

2.1 Network Modeling

Computation offloading modeling refers to the process of developing mathematical and computational models that serve to capture behavior, decision-making and performance aspects of computation offloading systems. It plays a crucial role in analyzing and optimizing computation offloading in Mobile Edge Computing [17]. Moreover, the process of computation offloading modeling depends heavily on some components, such as task characteristics, offloading decision, channel model and computation resource allocation.

In the context of Mobile Edge Computing, both network modeling and channel modeling are important to understand and optimize the system's performance. While network modeling provides a general view of the system's structure and connectivity, channel modeling focuses specifically on wireless communications characteristics and optimization.

The design of MEC networks addresses two aspects of network performance: wireless communication and edge computing. While the wireless communication aspect focuses on optimizing the transmission of data and tasks between mobile devices and edge servers, the edge computing aspect revolves around bringing computational resources closer to the network edge, which enables low-latency and high bandwidth processing of tasks and data.

In our paper, we are going to shed the light on wireless communication, which involves channel allocation, resource allocation, and transmission protocols to ensure reliable and efficient wireless communication by considering factors such as signal strength, interference and channel capacity. Measuring signal quality can be defined as the ratio of wanted signal strength and the unwanted interference plus noise, it is known as Signal to Interference Noise Ratio (SINR) which is the difference in decibels between the received signal and the background noise level, and is calculated as follows:

$$SINR = \frac{S}{N} \tag{1}$$

where S (watts) is the average received signal power over the bandwidth, N (watts) is the average power of the noise and interference over the bandwidth. The network operator aims to maximize SINR to deliver the best user experience, either by transmitting at a higher power, or by minimizing the interference and noise, generally, a SINR value that is more than 25 dB is considered excellent when measuring the quality of a wireless connection. In theory, the channel capacity can be expressed as:

$$C = bw \log_2(1 + \frac{S}{N}) \tag{2}$$

where C is the channel capacity, bw is the bandwidth of the channel and S/N represents the SNIR value as mentioned above.

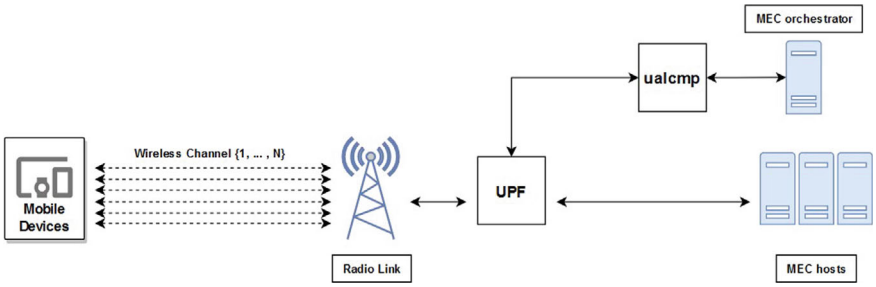


Fig. 1 Cellular network architecture

Channel allocation, being a critical aspect of efficient wireless communication,will be a key focus of our study. We will examine the complexities and limitations involved in allocating channels for transmitting data between devices, taking into account factors such as channel capacity, interference, and available resources. The channel allocation problem requires careful consideration to ensure optimal utilization of the wireless spectrum and efficient data transmission, the authors in [18] define channel allocation as the process of choosing the spectrum range for the offloading of computational tasks. Its process typically involves assigning communication channels or frequencies to nodes in a network, and it can be done based on various channel allocation algorithms and policies according to the requirements of the simulation scenario.

Our adopted topology is illustrated by the following Fig. 1, in which we integrate in the MEC system a set of MEC hosts, a MEC orchestrator that receives requests from user devices via the User Application Lifecycle Management Proxy (UALCMP). The MEC hosts serve as distributed computational nodes, positioned near the edge network to reduce latency and enhance the overall system performance. The MEC orchestrator acts as the central controller, managing resource allocation, load balancing and task scheduling, to ensure optimal utilization of the available resources. The UALCMP is the intermediary that transmits user requests to the MEC orchestrator and manages the life cycle of the applications, from deployment to termination. In addition to the MEC system level, the core network level plays an important role in enabling edge computing capabilities. On this level, we find a critical component called User Plane Function (UPF), responsible for managing the user data traffic.

In this architecture, we have multiple mobile devices running applications. Those applications request instantiation from the UALCMP, the MEC orchestrator selects a suitable host based on the needed requirements to run the application.

2.2 Channel Allocation Problem

The channel allocation problem in MEC arises due to the limited availability of wireless spectrum resources and the need to ensure optimal utilization of these resources in a dynamic and heterogeneous network environment in which we find multiple users sharing and accessing the same spectrum resource. Therefore, it specifically focuses on optimizing the allocation of communication channels within MEC networks to facilitate efficient and reliable data transmission between mobile devices and edge servers by minimizing interference, maximizing network capacity and ensuring a fair and efficient utilization of the available channels.

Channel allocation strategies are formulated to ensure the effective utilization of frequencies, time slots, and bandwidth. There are various approaches to tackle the channel allocation problem, such as centralized approaches where a base station manages the allocation, and distributed approaches where devices collaborate to allocate channels.

Machine learning algorithms, optimization algorithms, and game-theoretic models can be utilized to optimize channel allocation decisions based on factors such as channel quality, traffic load, and user requirements. These schemes play a critical role in optimizing network performance, mitigating interference, and improving the overall system capacity. Here are some common channel allocation schemes:

- Fixed Channel Assignment: in this case, channels are statically assigned to mobile devices based on a predetermined path, this approach is simple and easy to implement, however it may lead to non optimal resource utilization.
- Dynamic Channel Allocation: channels are dynamically reassigned based on changing network conditions in order to adapt to the variations of user demand, interference levels and network topology. Channel hopping is considered one of the algorithms that fall under this category. The authors in [19] proposed a communication resources allocation algorithm that selects the best communication path between the mobile device and the base station where the virtual machine is deployed.
- Load-based channel allocation: this approach refers to the dynamic assignment of wireless communication channels to devices based on the current load or traffic conditions in the MEC network, aiming for efficient resource utilization, interference minimization and overall network performance improvement.
- Game-Theoretic approaches: game theory models can be employed to address channel allocation as a strategic interaction among different devices, aiming to achieve a Nash equilibrium that optimizes overall network performance [20, 21].
- Machine learning-based allocation: this approach relies on techniques such as reinforcement learning and deep learning, it can be applied to optimize channel allocation dynamically based on real-time data, network conditions and historical patterns. It is characterized by its adaptability, intelligent decision-making and potential self-optimization.

3 Simulation and Results

Based on the described architecture above in Fig. 1, we conducted an experiment where three mobile devices, moving linearly, offload their workload to an edge server that contains multiple MEC hosts, passing by the UALCMP that forwards the user's requests to a MEC orchestrator in order to assign the workload to a suitable MEC host for processing.

The nodes are set in a playground area where X = 1100 m, Y = 800 and Z = 50 m. In this simulation, the mobility is taken into consideration, and each equipment has a specific configuration. We set properties for running application in the mobile device, the MEC system as well as the channel model.

The characteristics of the MEC host are shown below in Table 1:

Additionally, Mobile Edge Computing is considered as the key enabler of the $5th$ Generation networks (5G) [22]. 5G relies on NR technology to provide an enhanced mobile broadband, a reliable low latency communication and the ability to support a massive number of connected devices.

The modeling of our channel is based on the New Radio (NR) technology with *20* mW maximum sending power, -110 dBm of signal attenuation threshold, a path loss coefficient equals to *2*, a *2.4* GHz carrier frequency and a *Freespace Model* in terms of propagation. The simulation is 38 seconds long, the user devices move linearly across the playground by a 10 mps (meters per second) speed, each starting from an initial (X,Y) placement. Moreover, we adopted carrier aggregation by using 25 frequency bands in a single component carrier.

Aiming to test the adopted wireless communication, SINR is used to measure the quality of the connection, in Fig. 2, we recorded the SINR value at the Downlink and Uplink of the wireless link between the 1st user and the serving base station.Knowing that maximizing SINR delivers a good quality of service, distance plays a major role in degrading it, since the link is more exposed to noise and interference.

Figure 2 is a representation of SINR measurements of the three mobile user devices, whether on the Downlink or the Uplink, accordingly with their distance to the serving base station. We notice that the degradation of the SINR is related to the growing distance resulting in having a poor communication quality.

Below, Table 2 shows the different SINR values and their correspondent signal quality.

It is also important to note, as shown in Fig. 4, that the mean response time of the MEC host is *0.000216* s, meanwhile the UALCMP responds in a mean of *0.000014* s.

Table 1 MEC host characteristics

Maximum running applications	100
RAM	32 GB
Storage	100 TB
CPU speed	400000 MIPS

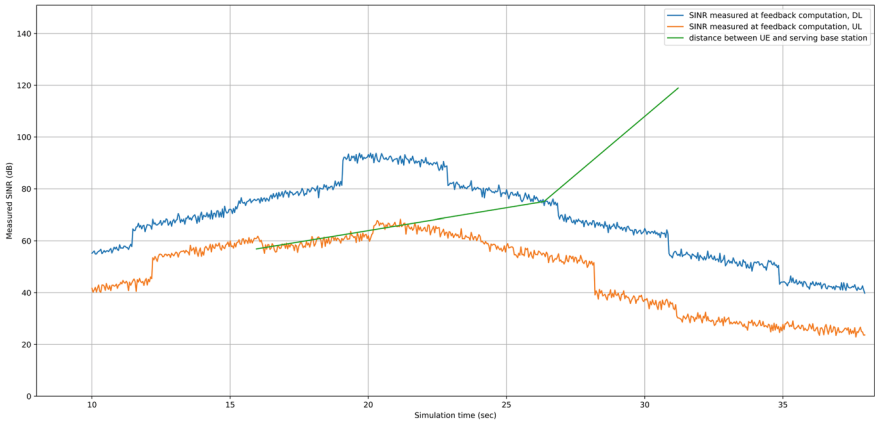


Fig. 2 Measured Downlink and Uplink SINR in each user equipment accordingly with the distance

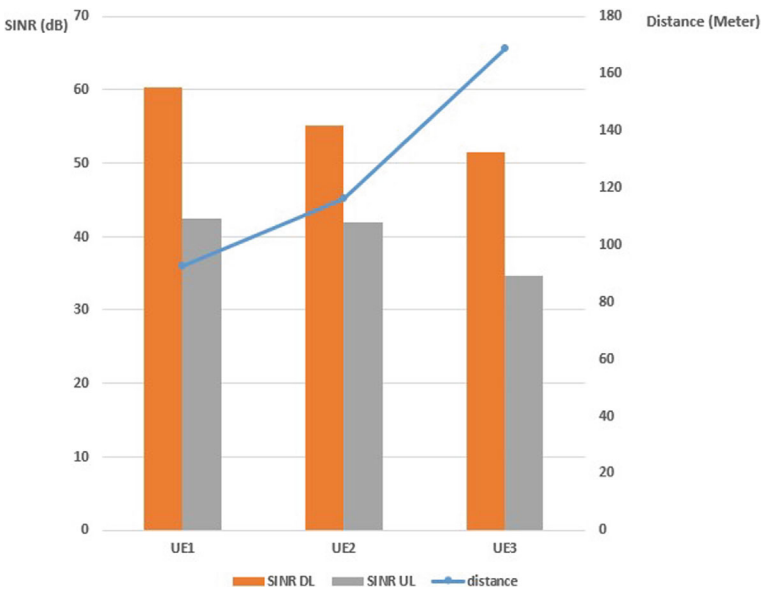


Fig. 3 Measured Downlink and Uplink SINR at computation feedback

Table 2 SINR quality

Radio frequency condition	SINR (dB)
Excellent	≥20
Good	13 to 20
Mid cell	0 to 13
Cell edge	≤0

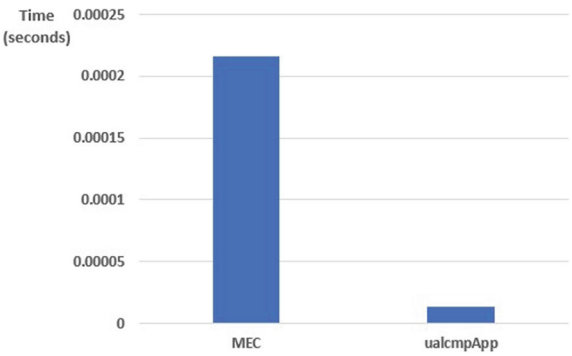


Fig. 4 Response time

Those results explain that even though latency can impact computation offloading, the processing time also plays a major role in having a real time response.

4 Conclusion and Future Work

The aim of this paper is to study the impact of constraints such as interference, noise and distance on the channel quality. We conducted an experiment where three mobile devices use the same channel to offload their workload to a MEC host, the results have shown that factors such as channel quality, distance and user requirements are important to model an optimized channel allocation approach. While this is a first attempt to study the channel allocation problem, our purpose resides in producing a more elaborated contribution.

References

1. Kumar, K., Lu, Y.: Cloud computing for mobile users: can offloading computation save energy? Computer **43**, 51–56 (2010)
2. Mach, P., Becvar, Z.: Mobile edge computing: a survey on architecture and computation offloading. IEEE Commun. Surv. Tutorials **19**, 1628–1656 (2017)
3. Maftah, S., EL Ghmary, M., El Bouabidi, H., Amnai, M., Ouacha, A.: Optimal resource allocation in mobile edge computing based on virtual machine migration. International Conference On Big Data and Internet of Things, pp. 575–585 (2022)
4. Ouacha, A., El Ghmary, M.: Virtual machine migration in mec based artificial intelligence technique. IAES Int. J. Artif. Intell. **10**, 244 (2021)
5. Maftah, S., El Ghmary, M., El Bouabidi, H., Amnai, M., Ouacha, A.: Optimal task processing and energy consumption using intelligent offloading in mobile edge computing. Int. J. Interactive Mobile Technol. **16** (2022)
6. Chen, X., Zhang, H., Wu, C., Mao, S., Ji, Y., Bennis, M.: Performance optimization in mobile-edge computing via deep reinforcement learning. 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall). pp. 1–6 (2018)

7. El Ghmary, M., Hmimz, Y., Chanyour, T., Malki, M.: Energy and processing time efficiency for an optimal offloading in a mobile edge computing node. Int. J. Commun. Netw. Inf. Secur. **12**, 389–393 (2020)
8. Sadatdiynov, K., Cui, L., Zhang, L., Huang, J., Salloum, S., Mahmud, M.: A review of optimization methods for computation offloading in edge computing networks. Digital Commun. Netw. (2022)
9. Gilly, K., Bernad, C., Roig, P., Alcaraz, S., Filiposka, S.: End-to-end simulation environment for mobile edge computing. Simul. Model. Pract. Theory **121**, 102657 (2022)
10. Hasanin, T., Alsobhi, A., Khadidos, A., Qahmash, A., Khadidos, A., Ogunmola, G.: Efficient multiuser computation for mobile-edge computing in IoT application using optimization algorithm. Appl. Bionics Biomech. **2021**, 1–12 (2021)
11. Gao, W., Zhao, Z., Yu, Z., Min, G., Yang, M., Huang, W.: Edge-computing-based channel allocation for deadline-driven IoT networks. IEEE Trans. Indus. Inf. **16**, 6693–6702 (2020)
12. Wang, C., Liang, C., Yu, F., Chen, Q., Tang, L.: Computation offloading and resource allocation in wireless cellular networks with mobile edge computing. IEEE Trans. Wirel. Commun. **16**, 4924–4938 (2017)
13. Zhang, D., Piao, M., Zhang, T., Chen, C., Zhu, H.: New algorithm of multi-strategy channel allocation for edge computing. AEU-Int. J. Electron. Commun. **126**, 153372 (2020)
14. Zhang, X., Mao, Y., Zhang, J., Letaief, K.: Multi-objective resource allocation for mobile edge computing systems. 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), pp. 1–5 (2017)
15. Zeng, M., Hao, W., Dobre, O., Poor, H.: Delay minimization for massive MIMO assisted mobile edge computing. IEEE Trans. Veh. Technol. **69**, 6788–6792 (2020)
16. Ren, Y., Guo, A., Song, C., Xing, Y.: Dynamic resource allocation scheme and deep deterministic policy gradient-based mobile edge computing slices system. IEEE Access. **9**, 86062–86073 (2021)
17. Lin, H., Zeadally, S., Chen, Z., Labiod, H., Wang, L.: A survey on computation offloading modeling for edge computing. J. Netw. Comput. Appl. **169**, 102781 (2020)
18. Yang, J., Shah, A., Pezaros, D.: A survey of energy optimization approaches for computational task offloading and resource allocation in MEC networks. Electronics **12**, 3548 (2023)
19. Plachy, J., Becvar, Z., Strinati, E., Pietro, N.: Dynamic allocation of computing and communication resources in multi-access edge computing for mobile users. IEEE Trans. Netw. Serv. Manag. **18**, 2089–2106 (2021)
20. Cui, G., He, Q., Chen, F., Zhang, Y., Jin, H., Yang, Y.: Interference-aware game-theoretic device allocation for mobile edge computing. IEEE Trans. Mobile Comput. **21**, 4001–4012 (2021)
21. Chu, S., Fang, Z., Song, S., Zhang, Z., Gao, C., Xu, C.: Efficient multi-channel computation offloading for mobile edge computing: A game-theoretic approach. IEEE Trans. Cloud Comput. **10**, 1738–1750 (2020)
22. Pham, Q., Fang, F., Ha, V., Piran, M., Le, M., Le, L., Hwang, W., Ding, Z.: A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art. IEEE Access. **8**, 116974–117017 (2020)

Optimization Strategies in Mobile Edge Computing Through Intelligent Task Offloading

Nouhaila Moussammi, Mohamed El Ghmary, and Abdellah Idrissi

Abstract With the rapidly expanding use of mobile devices and the increasing importance of real-time data processing, the need for efficient and effective solutions to handle data and improve network performance has become paramount. Mobile Edge Computing (MEC) has emerged as a promising solution, decentralizing computational resources closer to end-users, thereby mitigating computing delays and network congestion. A pivotal facet of MEC is computation offloading, which empowers mobile devices to transfer computational tasks to MEC servers equipped with superior computing capabilities, thus reducing latency and preserving device battery life. This study aims to explore and address the trade-off between energy usage and latency that naturally exists within MEC networks. It takes into account the limitations imposed by restricted energy resources while actively working towards the reduction of latency. The research introduces a comprehensive algorithmic model for energy-aware offloading and defines a set of criteria for assessing its performance. Furthermore, the study conducts an in-depth examination, classification, summary, and comparative analysis of existing offloading algorithms, all guided by the proposed algorithmic and evaluation criteria. The findings demonstrate highly favorable trade-offs between execution time and energy consumption. This enhances the efficacy of offloading while simultaneously lowering energy usage and execution time.

Keywords Mobile edge computing · Intelligent task offloading · Energy efficiency

N. Moussammi (✉) · A. Idrissi (✉)
Artificial Intelligence & Data Science Group, IPSS Team, Faculty of Science of Rabat, Mohammed V University, Rabat, Morocco
e-mail: nouhaila_moussammi@um5.ac.ma

A. Idrissi
e-mail: a.idrissi@um5r.ac.ma

M. El Ghmary (✉)
Department of Computer Science, FSDM, Sidi Mohamed Ben Abdellah University, Fez, Morocco
e-mail: mohamed.elghmary@usmba.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_43

1 Introduction

The internet has brought about a tremendous increase in the amount of data being generated, and this data requires more and more computational resources to process in real-time. Cloud computing, with its centralized access to computational resources, has become a popular solution for dealing with this growing demand. However, the high volume of data and limitations in transmission channels can put a significant strain on networks, leading to delays and congestion, which can negatively impact the customer experience [1]. To address these challenges, a new computing paradigm, Mobile Edge Computing (MEC) [2], has been developed to offer a range of advantages, benefits, and motivations for its use. One of the key advantages of MEC is its ability to provide localized computing power through the use of resources available at the edge of the network, also known as edge cloud [3].

By using MEC, users can access and utilize edge cloud resources in close proximity, eliminating the need for long-distance data transmission to a remote central cloud, thus providing a more efficient, cost-effective, and low-latency data processing experience. The use of MEC servers enables data processing tasks to be executed on edge devices or edge cloud resources, thereby reducing the transmission delays and costs associated with sending data to a remote central cloud for processing. This can lead to significant improvement in system performance and faster processing of data.

Another key benefit of MEC is its ability to reduce the pressure on the backhaul network, which can lead to improved overall system performance. MEC's ability to perform computation at the edge can also reduce energy consumption, as data does not need to be transmitted over long distances, and this can be particularly useful for IoT devices which have limited battery life. MEC also presents new opportunities for providing low-latency services, which are essential for emerging technologies such as virtual and augmented reality. MEC's ability to provide low-latency and low-power communication makes it an ideal solution for supporting the use of IoT devices. Additionally, MEC can play a crucial role in supporting the use of 5G networks, which have a high demand for low-latency, high-bandwidth services. In summary, MEC provides a range of advantages, benefits, and motivations for its use. It is an efficient, cost-effective, and low-latency alternative to traditional cloud computing and is expected to become a key technology in the future [4]. This paper delves into the topic of MEC and offloading computational tasks. The second section of this paper outlines the procedure for utilizing MEC for offloading computational tasks. In the third section, explains the system model and its formulation. The fourth section, the strengths and limitations of current algorithms used for offloading. the fifth section examines an implementation and Evaluation. Ultimately, the goal of this paper is to propose improvements to current offloading techniques. Finally the section five concludes the paper by summarizing the findings and contributions.

2 Procedure for Utilizing MEC for Offloading Computational Tasks

Before proceeding with computing offloading, the MEC server must go through a series of steps, as outlined in Fig. 1. One of the initial steps is to determine the legitimacy of users who are requesting this service. This step is crucial in ensuring that only authorized users are provided with access to the system's resources. If the user is determined to be legitimate, the MEC server will then allocate resources to them in a controlled and efficient manner. This includes evaluating the user's resource usage patterns, the amount of data associated with their task, and real-time updates on resource availability and system energy conditions. This helps to ensure that resources are used in the most efficient way possible, and that the system's performance is optimized for all users.

On the other hand, if the user is found to be illegitimate, the MEC server will deny service to that user. This is an important step in ensuring the security and integrity of the system, and it helps to prevent unauthorized access or misuse of the system's resources. It's worth to mention that the system should be designed to handle the allocation of resources in an efficient and fair manner to all users.

This may include using algorithms and other techniques to optimize resource allocation, and implementing policies that help to ensure that resources are used in an equitable and fair way.

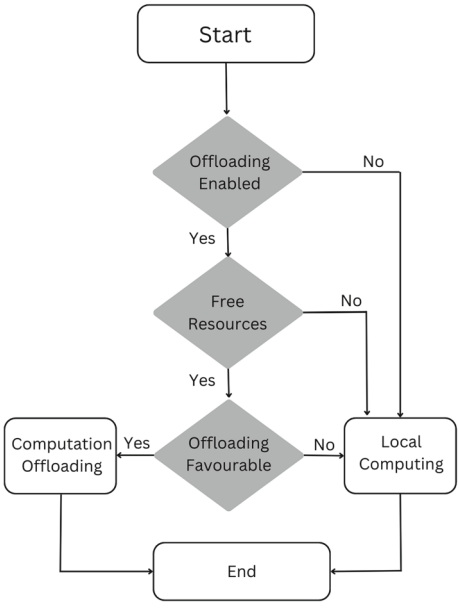


Fig. 1 Process of computing offloading for MEC

3 System Model and Problem Formulation

We presume that the layer of users is comprised of N smart mobiles devices (MDs) with $N = \{1, 2, ..., n\}$, and single MEC server.

A system is composed of K tasks with $K = \{1, 2, ..., k\}$. The tuple named $K_i = (C_i, S_c, D_i, d_{t_i})$ comprises of the number of CPU cycles needed for an operation to be completed that corresponds to the task (C_i).

The tuple includes the offloading of the source code of IoT to MEC (S_c) and the data input associated with every task (D_i) and also as the deadline of the completion (d_{t_i}).

To facilitate efficient communication, the network deploys a robust wireless infrastructure. Every scheduled task is endowed with a designated wireless channel, denoted by the set $A = \{a_1, a_2, ..., a_k\}$ These channels serve as the conduit through which data flows seamlessly, facilitating the exchange of information between the IoT devices and MEC servers (Fig. 2).

3.1 Computation on Local Devices

When it comes to computing, local computing refers to the process of handling data and processing it directly on the node device, without the involvement of the MEC

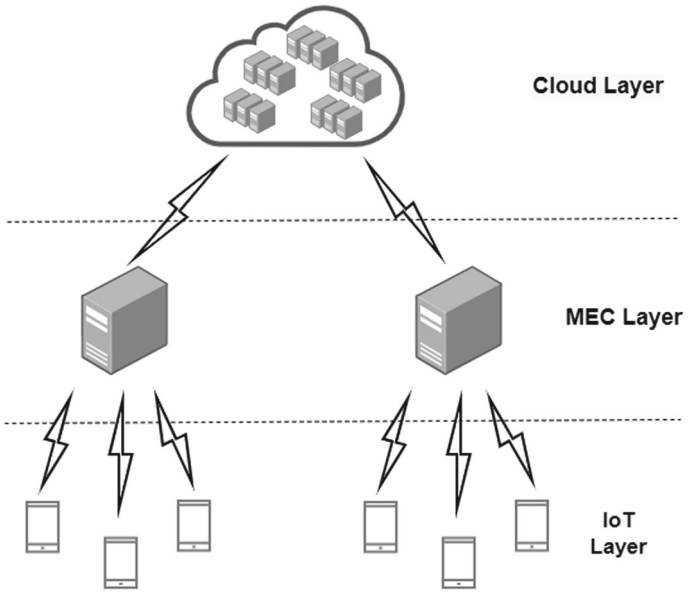


Fig. 2 System model of MEC

server. In many cases, mobile devices are equipped to handle data processing locally. However, the efficiency of local computing can be influenced by a number of factors, such as the performance of the CPU and the speed at which data can be read.

A device's Central Processing Unit (CPU) performance plays a vital role in the efficiency of local computing. The CPU is responsible for performing various computational tasks, such as arithmetic operations, logical operations and data transfer. A device with a high-performance CPU can process more data in less time, resulting in a better local computing experience.

Additionally, the read speed of data can also affect local computing. The faster data can be read from the storage, the faster it can be processed by the CPU, and the more efficiently local computing can be done. However, it's important to note that local computing also has its own limitations. Depending on the nature of the task and the resources available on the device, it might not be able to handle the computation efficiently or even not possible to complete the task. In these cases, it's better to offload the data to the MEC server to ensure better performance. In our System model, the processing time for task i when executed locally is denoted as T_i^{loc}, and it's determined by Eq. 1:

$$T_i^{loc} = \frac{C_i}{f_j^{loc}} \tag{1}$$

The energy consumption for each mobile device is determined by Eq. 2:

$$E_i^{loc} = T_i^{loc} * P_i^{loc} \tag{2}$$

Where P_i^{loc} is the local power consumption rate for the MDs i. The local cost of one task i is expressed as follows:

$$Cost_i^{loc} = y_{iT} * T_i^{loc} + y_{iE} * E_i^{loc} \tag{3}$$

Where $y_{iT} \in \{0, 1\}$ and $y_{iE} \in \{0, 1\}$ represents the weightings and $y_{iT} + y_{iE} = 1$. They also represents a trade-off between execution time and energy consumption and minimize one of the costs.

3.2 Complete Offloading

Complete offloading, involves sending all task data to the MEC server, where it is processed using the MEC server's computing resources. This approach allows devices with limited resources to offload their computational tasks to a more powerful server. When using complete offloading, the total time to complete a task can be broken down into two main components: the time taken for data transmission and the time taken for data processing.

The effectiveness of complete offloading is dependent on a number of factors. One of the key considerations is the capacity of the channel that is used for data transmission. A higher capacity channel will allow for faster data transfer, which will in turn reduce the overall task completion time. The quality of the channel is another factor that affects the effectiveness of complete offloading. A channel that experiences high levels of noise or interference can negatively impact the data transmission time. Another important factor is the computing power of the MEC server. A server with more powerful computing resources will be able to process data faster, which in turn will reduce the data processing time and improve the overall performance of the system.

The uplink transmission data rate (R) is a critical factor that governs the speed at which data can be transmitted from a mobile device to a server Edge over the communication channel. This data rate is expressed in bits per second (bps) and plays a crucial role in determining the overall efficiency of task offloading and execution.

$$R = B * log_2(\frac{h^2 * p}{\sigma} + 1) \tag{4}$$

The uplink data rate (R) is determined by considering several key parameters that influence the performance of the communication channel.

Bandwidth (B) represents the available bandwidth for the communication channel.

The channel gain (h) represents the signal power between the mobile device and the server, reflecting the strength of the transmitted signal. A higher channel gain indicates a stronger and more reliable connection. The transmission power (p) denotes the rate at which the user needs to transmit data to process input data onto an Edge server located nearby. The transmission power is an essential factor in determining the data transfer rate. The background noise (σ) accounts for any unwanted interference or disturbance in the communication channel. A lower background noise level results in a more favorable signal-to-noise ratio.

$$T_i^{mec} = T_i^{up} + T_i^{exe} + T_i^{res} \tag{5}$$

Where: $T_i^{up} = \frac{S_c + D_i}{R}$, $T_i^{exe} = \frac{C_i}{F^{mec}}$ and T_i^{res} represents the time required to resend the results of the calculation to a user after the task has been executed. It is important to note that this time is generally ignored in the algorithm, and it is in accordance with the considerations mentioned in the reference [5].

The calculation of energy consumption for data communications from the local MEC server is determined using Eq. (6). This equation takes into account the energy taken for data transmission E^{up} and the energy expended during task execution E^{exe}. Similarly, the dynamic energy consumed by the MEC server is evaluated using a similar approach as that used for the IoT device.

$$E_i^{mec} = E_i^{up} + E_i^{exe} = T_i^{up} * P_i^{up} + T_i^{exe} * P_i^{exe} \tag{6}$$

Where P_i^{up} and P_i^{exe} represent the energy consumption rate per unit time during data transmission (upload) and task execution. The cost for the mec server is expressed as follows:

$$Cost_i^{mec} = y_{iT} * T_i^{mec} + y_{iE} * E_i^{up} \tag{7}$$

3.3 Partial Computation on MEC Server

Partial offloading is a technique for managing computational tasks that involves splitting the task into two parts—one portion executed on the device and another portion on the mobile edge computing (MEC) server. The scheduling of data is determined by an offloading algorithm. When using partial offloading, several factors must be taken into consideration to optimize performance and resource usage. These include transmission time, processing time, energy consumption and resource allocation. Each of these factors affects the overall system efficiency and interacts with one another, meaning changes to one factor can impact the entire system's performance.

One of the key advantages of partial offloading is that it enables more efficient use of resources. By transferring only those tasks that require more computational power or storage to the MEC server, and processing the remaining tasks locally, it ensures a balance between utilizing local resources and offloading tasks to the MEC server. This balance can lead to improved system performance.

Additionally, partial offloading enables dynamic offloading where the decision of offloading can be taken at runtime based on the network conditions, device capabilities and energy constraints. This enables more efficient use of resources and better performance. However, it's worth noting that partial offloading is considered the most complex model among the three offloading methods, since it requires the integration of a multitude of influencing factors and the decision of which data to be processed locally and which one to be offloaded must be taken carefully, taking into account the specific requirements of the application, the available resources and the trade-offs involved.

3.4 Evaluation Parameters for Computing Offloading

When evaluating the performance of an offloading algorithm, there are several key parameters to consider, such as time delay, bandwidth utilization, and applicability. These three aspects are crucial to understand in order to optimize the performance of the offloading process.

1. Time delay refers to the total time taken for a system to respond to a user or application's request for computing services. This includes both the response time and the data processing time. Factors that can impact the length of time delay include the efficiency of the offloading algorithm, channel capacity, bandwidth, and the amount of task data. Depending on the scenario, the time delay of different offloading models can also vary. For example, in real-time applications such as online games or video interactions, a shorter time delay is required than in non-real-time applications like data caching.

2. Bandwidth utilization is another important aspect to consider during the offloading process. A significant amount of data is transferred, which can lead to channel congestion and result in high time costs. With limited resources during wireless transmission, even though 4G technology can improve wireless transmission performance, the downsides of wireless transmission cannot be entirely eliminated. As a result, increasing bandwidth utilization can improve the resource utilization of offloading, thus making it an important factor that affects the effectiveness of complete offloading.

3. Algorithm applicability is also crucial to consider when implementing offloading. Different applications have different computational resource and energy needs. For example, online games and video interactions require shorter time delays than data caching, and nodes processing large amounts of data consume a lot of energy. Therefore, it is not possible for a one-size-fits-all offloading model to apply to all scenarios. The optimal model can be determined by carefully assessing the characteristics and needs of the user from all perspectives. This includes taking into account the computational requirements of the application, the available resources on the device and the network conditions.

4 An Examination of Current Offloading Algorithms

4.1 Algorithms that Minimizes the Execution Time

In the study by Liu et al. [6], a Markov decision process is employed to optimize the integration of various factors, such as the status of computing missions, local resource utilization, wireless transmission channel utilization, and mobile edge computing (MEC) server utilization, for efficient computation offloading. By analyzing data processing time and energy costs, the algorithm aims to minimize execution time while satisfying constraints. This approach significantly decreases execution time when compared to not utilizing offloading, but it is also quite complex. This article [7] proposes an AI-based approach, using the ant colony algorithm, to migrate virtual machines in a mobile edge computing environment. Simulations in NS3 confirm its effectiveness in adapting to resource-constrained mobile devices. On the other hand, Mao et al. [8] proposed a MEC system that utilizes green energy harvesting

[9] to decrease traditional energy consumption. The algorithm employs Lyapunov optimization to maintain system stability, while considering the CPU's periodic frequency and offloading power in decision-making. Real-time resource allocation is implemented through binary search. This algorithm is beneficial as it utilizes green energy and optimizes execution time, but it is not widely applicable as it fully offloads data.

4.2 Algorithms that Minimizes the Energy Consumption

Shan et al. [10] proposed a resource allocation algorithm that differentiates between online and offline cases, using Markov chain to allocate computing resources and energy. This approach allows for more efficient use of resources by considering the current conditions and the needs of the system, rather than assuming fixed resource allocation. In this way, it can adjust the resources dynamically based on the current situation, resulting in energy savings. However, the algorithm has its limitation as it only focuses on the energy efficiency, but does not consider other factors such as time delay and data processing efficiency. Additionally, the research conducted by Shi et al. [11] proposed an offloading algorithm for distributed mobile cloud computing environments. By analyzing the movement patterns of nodes, the algorithm developed a network access prediction system based on tail matching of sub-sequences, providing a new approach to offloading in mobile cloud computing This algorithm is designed to improve the energy efficiency of the system, by taking into account the dynamic nature of mobile environments, and predicting the access patterns of the nodes in order to optimally allocate resources. Its major advantage is that it is suitable for many scenarios and saving energy. However, it also has a limitation as it does not consider the implementation of time delay and data processing efficiency and mobile perception is not always stable, thus their algorithm does not take into account the instability of the mobile perception. The studies by [12–14] highlights the pivotal role of offloading decisions, radio resource allocation, and local computational resource optimization in a multi-user MEC system. Introducing the Overall Energy Minimization by Resources Partitioning (OEMRP) heuristic scheme, the research successfully minimizes overall energy consumption, ensuring prolonged battery lifetime for smart mobile devices. In our previous work, we proposed a multi-user system [12] and a single-user system [15], both of which include multiple tasks and high-density computing in order to minimize energy consumption.

4.3 The Proposed Algorithm: Energy-Time Trade-Off Task Offloading

In this analysis, we examine various algorithms for computing offloading, specifically focusing on the efficiency and effectiveness of each approach. One algorithm, proposed by Chen et al. [16], prioritizes energy conservation by making decisions

on task data transfer based on the current state of the system. While this method can save up to 50% energy compared to non-offloading solutions, it does not consider the energy and time consumption of data transmission. Another approach, proposed by Mao et al. [17], addresses the issue of channel assignment in wireless transmission using Gaussian method and Lyapunov optimization for resource and energy allocation. However, it does not take into account interference and the impact of data transmission time on performance. This article [18] focuses on optimizing Mobile Edge Computing for smart devices, introducing a heuristic for efficient computation offloading and demonstrating promising results in treatment time and energy consumption.

Lyu et al. [19] propose an algorithm that utilizes information from the IoT environment and optimizes feedback to the MEC server for energy balance and system stability. This approach reduces the number of nodes sending feedback, but may not always result in the most efficient offloading decision. Liu et al. [20] implements a multi-variable optimization algorithm for MEC offloading, analyzing execution time, energy consumption, and unit energy yield. But it doesn't take into account the local computing offloading and the total data offloading probability doesn't help with real-time offloading decisions. Meng-Hsi et al. [21] propose a multi-user data offloading algorithm that jointly optimizes computation delay, energy consumption, and total cost of computing. But the joint optimization result doesn't take into account the running time cost.

Lastly, Cao et al. [22] uses game theory to make offloading decisions for multi-user scenarios but it assumes all users have the same computing power and data amount which is not common in reality and thus the algorithm is not very general. Overall, these algorithms aim to reduce energy consumption in partial offloading for multi-user scenarios, but further improvements can be made to address the overall effect of consumption or execution time delay.

Algorithm 1 Energy-Time Trade-off Task Offloading Algorithm

Input: Collection of data sets of mobile devices and MEC server and Initialization.
Output: A mapping of tasks to their offloading options (Local, Edge) .

for $j \in$ (list of MDs) **do**
  Sort Task List based on task deadlines
  for $i \in$ (List of Tasks) **do**
    Calculate energy and delay for local execution
    Calculate energy and delay for offloading
    Calculate the cost that balances energy consumption and execution time
    if $Cost <= 1$ or ($E_i^{loc} < E_i^{mec}$ and $T_i^{loc} <= d_t$) **then**
      Offloading Decision Task = 0
    else
      Offloading Decision Task = 1
    end if
  end for
end for
Return Offloading Decision, Energy, Delay.

The presented task offloading algorithm is designed for mobile edge computing (MEC) environments, where mobile devices can offload computational tasks to nearby edge servers. The primary goal of this algorithm is to make intelligent

decisions regarding task offloading, with a focus on optimizing energy consumption and execution time. The algorithm operates as follows:

Initialization: Mobile devices in the MEC system are initialized with unique processing power and weighting factors. The weighting factor determines the balance between energy consumption and execution time for each device.

Task Offloading: For each mobile device and a given set of tasks, the algorithm simulates task execution scenarios. It calculates the energy and delay associated with both local execution and offloading the task to an edge server. A cost metric is computed for each task, considering the device's weighting factor. The algorithm makes a decision to execute the task either locally or by offloading it based on the cost metric and a predefined threshold value. The primary goal of this algorithm is to enhance the efficiency and effectiveness of task offloading in MEC environments. It achieves this by intelligently determining whether a task should be executed locally on a mobile device or offloaded to an edge server. The overarching objectives are as follows:

Optimizing Energy Consumption: The algorithm seeks to minimize energy consumption by offloading tasks when it results in lower energy usage compared to local execution.

Minimizing Execution Time: Simultaneously, the algorithm aims to minimize task execution time by favoring local execution when it is faster than offloading.

Balancing Trade-offs: The algorithm strikes a balance between energy consumption and execution time, considering the preferences set by the weighting factors of individual mobile devices.

Computational Complexity: The computational complexity of this algorithm depends primarily on the number of mobile devices (N) and the number of tasks simulated (K). In each simulation, for each device, tasks are generated and evaluated, resulting in a time complexity of approximately O(N * K). The overall complexity of the algorithm scales with the number of simulations and the complexity of task generation and evaluation within each simulation.

5 Implementation and Evaluation

In our study, we conducted a comparative analysis involving our newly proposed offloading approach (referred to as 'The proposed') and 'Local-Edge' representing the collaborative offloading decision as outlined in reference [23]. The experimentation involved a scenario with ten mobile devices operating within the same edge server domain. These devices generated a variable number of computation tasks, ranging from one to ten. As illustrated in Fig. 3a, it becomes evident that the execution time increases in direct correlation with the number of tasks.

When tasks were executed locally ('Local'), the execution time was notably higher compared to the other methods. This is primarily due to the limited computing capacity of mobile devices, rendering them unable to efficiently handle a substantial workload when performing tasks locally. In contrast, the offloading method we introduced

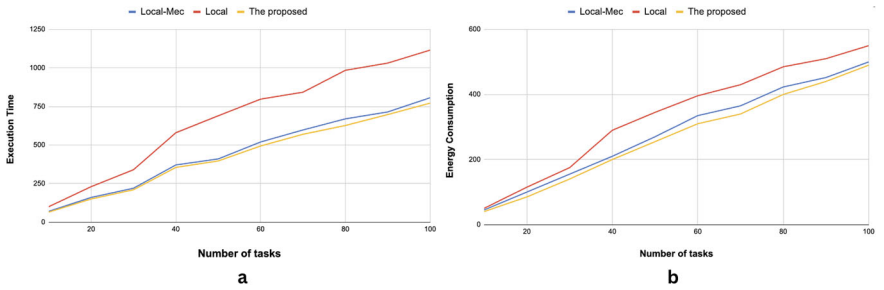


Fig. 3 Execution time and energy consumption of different methods

('The proposed') demonstrated the shortest execution times among all considered strategies. This outcome underscores the effectiveness of our proposed approach in achieving efficient task execution. As depicted in Fig. 3b, we observe a direct correlation between energy consumption and the number of tasks performed. Similarly to execution time, energy consumption experiences an increase as the task count rises.

Notably, when tasks are executed locally, the cumulative energy consumption reflects the usage of mobile devices alone. Local execution proves to be the most energy-intensive approach, potentially leading to excessive energy utilization by mobile devices. Such heightened energy consumption can not only negatively impact the user experience but also result in a shortened battery life for terminal devices. It's worth highlighting that our energy consumption calculations take into account both calculation and transmission energy expenditures associated with the tasks, excluding the device's intrinsic energy consumption

Each mobile device generates a total of 10 tasks, the distribution of tasks under different allocation schemes is depicted in Fig. 4. A notable trend emerges as the number of tasks generated by mobile devices increases—the proportion of tasks assigned to terminal devices decreases. This phenomenon arises from the inherent

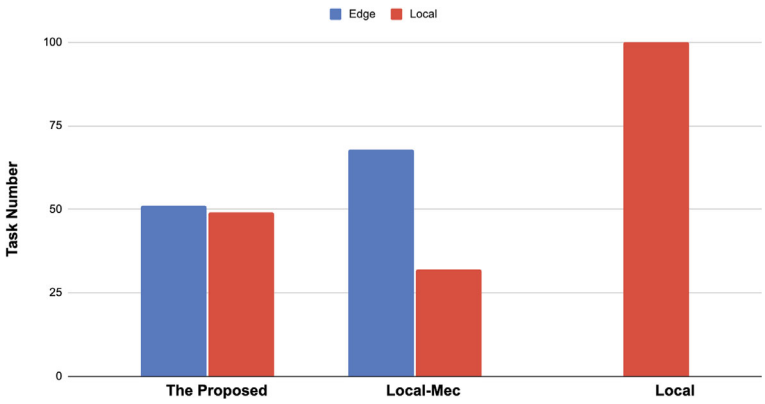


Fig. 4 The distribution of task allocation

limitations of mobile devices in terms of processing capacity. When faced with an increasing number of tasks, mobile devices may struggle to meet stringent maximum delay requirements. Consequently, a greater number of tasks are redirected towards edge server for execution.

6 Conclusion

Offloading computation is a vital technique in edge computing, which can enhance battery efficiency, decrease latency and boost performance of applications. Nevertheless, the impact of computation offloading is influenced by numerous factors, resulting in many offloading attempts failing to meet their desired outcomes. The proposed task offloading algorithm demonstrates its utility in MEC environments by effectively balancing energy consumption and execution time. By leveraging the inherent characteristics of each mobile device and considering task-specific attributes, the algorithm contributes to optimizing task allocation decisions. Through extensive simulations and visualizations, this research sheds light on the trade-offs associated with task offloading and provides valuable insights for enhancing the efficiency of MEC systems. We may also introduce more Artificial Intelligence concepts into this area in the future, including techniques published in [24–40] as an example.

References

1. Chen, Y., Zhang, N., Zhang, Y., Chen, X., Wu, W., Shen, X.: Energy efficient dynamic offloading in mobile edge computing for internet of things. IEEE Trans. Cloud Comput. **9**(3), 1050–1060 (2019)
2. Abbas, N., Zhang, Y., Taherkordi, A., Skeie, T.: Mobile edge computing: a survey. IEEE Internet Things J. **5**(1), 450–465 (2017)
3. Liu, L., Chen, C., Pei, Q., Maharjan, S., Zhang, Y.: Vehicular edge computing and networking: a survey. Mobile Netw. Appl. **26**(3), 1145–1168 (2021)
4. Khan, W.Z., Ahmed, E., Hakak, S., Yaqoob, I., Ahmed, A.: Edge computing: a survey. Future Gen. Comput. Syst. **97**, 219–235 (2019)
5. Lin, X., Zhang, H., Ji, H., Leung, V.C.: Joint computation and communication resource allocation in mobile-edge cloud computing networks. In: IEEE International Conference on Network Infrastructure and Digital Content, pp. 166–171 (2016)
6. Liu, J., Mao, Y., Zhang, J.: Delay-optimal computation task scheduling for mobile-edge computing systems. IEEE International Symposium on Information Theory, pp. 1451–1455. IEEE (2016)
7. Ouacha, A., El Ghmary, M.: Virtual machine migration in mec based artificial intelligence technique. IAES Int. J. Artif. Intell. **10**(1), 244 (2021)
8. Mao, Y., Zhang, J., Letaief, K.B.: Dynamic computation offloading for mobile-edge computing with energy harvesting devices. IEEE J. Sel. Areas Commun., 3590–3605 (2016)
9. Güler, B., Yener, A.: Energy-harvesting distributed machine learning. In: 2021 IEEE International Symposium on Information Theory (ISIT), pp. 320–325 (2021)

10. Shan, X., Zhi, H., Li, P., Han, Z.: A survey on computation offloading for mobile edge computing information. In: 2018 IEEE 4th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing,(HPSC) and IEEE International Conference on Intelligent Data and Security (IDS), pp. 248–251 (2018)
11. Shi, Y., Chen, S., Xu, X.: MAGA: a mobility-aware computation offloading decision for distributed mobile cloud computing. IEEE Int. Things J. **5**(1), 164–174 (2017)
12. Moussammi, N., El Ghmary, M., Idrissi, A.: Multi-task multi-user offloading in mobile edge computing. Int. J. Adv. Comput. Sci. Appl. **13**(12) (2022)
13. Hmimz, Y. et al.: Computation offloading to a mobile edge computing server with delay and energy constraints. In: 2019 International Conference on Wireless Technologies, Embedded and Intelligent Systems (WITS), pp. 1–6. IEEE (2019)
14. Hmimz, Y. et al.: Energy efficient and devices priority aware computation offloading to a mobile edge computing server. 2019 5th International Conference on Optimization and Applications (ICOA). IEEE (2019)
15. Moussammi, N. et al.: Multi-task offloading to a mec server with energy and delay constraint. In: The International Conference on Artificial Intelligence and Smart Environment. Cham: Springer International Publishing, pp. 642–648 (2022)
16. Chen, M.H., Liang, B., Dong, M.: Joint offloading and resource allocation for computation and communication in mobile cloud with computing access point. In: IEEE INFOCOM Conference on Computer Communications, pp. 1–9 (2017)
17. Mao, Y., Zhang, J., Song, S.H., Letaief, K.B.: Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems. IEEE Trans. Wirel. Commun. **16**(9), 5994–6009 (2017)
18. El Ghmary, M., et al.: Energy and processing time efficiency for an optimal offloading in a mobile edge computing node. Int. J. Commun. Netw. Inf. Secur. **12**(3), 389–393 (2020)
19. Hua, M., Tian, H., Lyu, X., Ni, W., Nie, G.: Online offloading scheduling for noma-aided mec under partial device knowledge. IEEE Int. Things J. **9**(3), 2227–2241 (2021)
20. Liu, L., Chang, Z., Guo, X.J. et al.: Multi-objective optimization for computation offloading in mobile-edge computing. IEEE Symposium on Computers and Communications, pp. 832–837 (2017)
21. Chen, M.H., Dong, M., Liang, B.: Resource sharing of a computing access point for multi-user mobile cloud offloading with delay constraints. IEEE Trans. Mobile Comput. **17**(12), 2868–2881 (2018)
22. Cao, H., Cai, J.: Distributed multi-user computation offloading for cloudlet based mobile cloud computing: a game-theoretic machine learning approach. IEEE Trans. Veh. Technol. **67**(1), 752–764 (2017)
23. Cui, L., Xu, C., Yang, S., Huang, J., Lu, N.: Joint optimization of energy consumption and latency in mobile edge computing for internet of things. IEEE Int. Things J. **6**(3), 4791–4803 (2019)
24. Idrissi, A., Li, C.M.: Modeling and optimization of the capacity allocation problem with constraints. RIVF, pp. 107–116 (2006)
25. Idrissi, A., Yakine, F.: Multicast routing with quality of service constraints in the ad hoc wireless networks. J. Comput. Sci. **10**, 1839–1849 (2014). https://doi.org/10.3844/jcssp.2014.1839.1849
26. Elhandri, K., Idrissi, A.: Parallelization of Top-k algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2021)
27. Retal, S., Idrissi, A.: A multi-objective optimization system for mobile gateways selection in vehicular Ad-Hoc networks. Comput. Electr. Eng. **73**, 289–303 (2018)
28. Idrissi, A.: Some methods to treat capacity allocation problems. J. Theoret. Appl. Inf. Technol. **37**(2), 141–158 (2012)
29. Rehioui, H., Idrissi, A.: A fast clustering approach for large multidimensional data. Int. J. Bus. Intell. Data Mining (2017)

30. Idrissi. A.: How to minimize the energy consumption in mobile ad-hoc networks (2012). arXiv preprint arXiv:1307.5910
31. Idrissi, A., Elhandri, K., Rehioui, H., Abourezq, M.: Top-k and skyline for cloud services research and selection system. International Conference on Big Data and Advanced Wireless Technologies (2016)
32. Abourezq, M., Idrissi, A., Yakine, F.: Routing in wireless Ad Hoc networks using the Skyline operator and an outranking method. Proceedings of the International Conference on Internet of things and Cloud Computing (2016)
33. ElHandri, K., Idrissi, A.: Parallelization of algorithm through a new hybrid recommendation system for big data in spark cloud computing framework. IEEE Syst. J. **15**(4), 4876–4886 (2020)
34. Zegrari, F., Idrissi, A.: Modeling of a dynamic and intelligent simulator at the infrastructure level of cloud services. J. Autom. Mobile Roboti. Intell. Syst. **14**(3), 65–70 (2020)
35. Zankadi, H., Idrissi, A., Daoudi, N., Hilal, I.: Identifying learners' topical interests from social media content to enrich their course preferences in MOOCs using topic modeling and NLP techniques. Int. J. Educ. Inf. Technol. **28**(5), 5567–5584 (2023)
36. Belmouhcine, A., Idrissi, A., Benkhalifa, M.: Web classification approach using reduced vector representation model based on Html tags. J. Theoret. Appl. Inf. Technol. **55**(1), 137–148 (2013)
37. Abourezq, M., Idrissi, A.: A Cloud Services Research and Selection System. IEEE ICMCS (2014)
38. Laghrissi, A., Retal, S., Idrissi, A.: Modeling and optimization of the network functions placement using constraint programming. Proceedings of the International Conference on Big Data and Advanced Wireless T(2016)
39. Abourezq, M., Idrissi, A., Rehioui, H.: An amelioration of the skyline algorithm used in the cloud service research and selection system. Int. J. High Perform. Syst. Archit. **9**(2–3), 136–148 (2020)
40. Elhandri, K., Idrissi, A.: Comparative study of Top-k based on Fagin's algorithm using correlation metrics in cloud computing QoS. Int. J. Int. Technol. Secur. Trans. **10** (2020)

AI for Enhanced Optimal Modeling in Wind Energy and Hydraulic Storage Systems with Lagrangian Insights

Abderrahim Ouza, Mohamed El Ghmary, Ali Choukri, and Adil Khazari

Abstract Wind power generation is a complex logistical undertaking with significant economic and social implications. The objective of this study is to formulate a method for the optimal management of the intermittent aspects of wind energy. The initial step entails modeling the various characteristics of the problem components, utilizing advanced techniques like Mixed Linear Programming within a scientific framework. Next, a mathematical formulation of management system of the wind power energy associated with storage hydraulic systems using the Lagrangian relaxation method is conducted. Conclusively, the deployment of operational policies is executed to articulate a systematic strategy for implementation.

Keywords Mixed linear programming · Hydraulic storage · Optimization · Lagrangian relaxation

A. Ouza (✉)
Faculty of Science, Ibn Tofail Univesity, Kénitra, Morocco
e-mail: Abderrahim.ouza@uit.ac.ma

M. El Ghmary
FSDM Sidi Mohamed Ben Abdelah University, Fez, Morocco
e-mail: Mohamed.elghmary@usmba.ac.ma

A. Choukri
Faculty of Science, Ibn Tofail University, Kénitra, Morocco
e-mail: ali.choukri@uit.ac.ma

A. Khazari
National School of Commerce and Management, Sidi Mohamed Ben Abdelah University, Fez, Morocco
e-mail: adil.khazari@usmba.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_44

1 Introduction

Renewable energy sources play a pivotal role in establishing an environmentally sustainable energy supply. Among these, hydro power stands out as a crucial contributor, holding immense potential to alleviate dependence on fossil fuels and mitigate greenhouse gas emissions. However, the evolving landscape of wind energy development introduces planning, exploitation, and management challenges, further compounded by the dynamics of liberalized electricity markets.

The intermittency and uncertainty inherent in wind energy output necessitate innovative solutions for stabilization, with energy storage emerging as a cornerstone [1, 2]. This paper explores the multifaceted realm of energy storage, a critical component in future energy systems relying heavily on variable renewable resources [3].

As we delve into the optimization of energy consumer and production systems, a holistic approach to energy storage management becomes imperative [4]. This entails a comprehensive study of source characteristics, predictive modes, and real-time control strategies.

The complexity of this endeavor is underscored not only by the intricate nature of the problem but also by the compelling environmental, economic, and social benefits inherent in the adoption of optimized pump/turbine-operation strategies for energy consumption/production [5]. This research aims to contribute novel insights to the global discourse on the integration and optimization of renewable energy sources.

2 Related Works

The work distinguishes itself by addressing the non-convex nature of the primal problem, acknowledging the feasibility challenges it poses [6]. The primary objective revolves around profit maximization, considering active power cost, pumping costs, and constraints on hydro and wind power. The proposed model integrates pumping operations, turbine operations, and energy storage dynamics, ensuring stability and efficiency [6]. Constraints encompass power capacity, energy capacity, and operational limits of the hydro-wind system [7].

Recently, significant researches involve considering previous studies on wind energy challenges, such as [1], emphasizing the significance of energy storage [2]. The work aligns with holistic approaches [4] and delves into optimization strategies for energy systems [4]. Contributions from [8, 9] on Lagrangian techniques are acknowledged, highlighting advancements in handling non-convex optimization problems. The unique aspect of avoiding parallel turbine-pumping operations draws inspiration from [6, 10], emphasizing profit maximization in coupled wind-hydro systems.

In the realm of programming languages, advancements in linear integer programming and structural insights into constrained optimization [11] contribute to the optimization approach [11, 12]. Moreover, [8] discuss advancements in Lagrangian techniques for non-convex optimization problems. [9, 13] propose multiplier updating methods for improved Lagrangian relaxation in energy systems optimization. These recent references enrich the study's scientific foundation, reflecting the current landscape of renewable energy research.

3 Formulation of the Problem

Over the years, the publications of the eolian combines the hydraulic storage management optimization were postponed. This work introduces advancements to the original Lagrangian relaxation technique through the incorporation of augmented Lagrangian methods and multiplier updating strategies. The efficacy of these techniques lies in computing optimal values for Lagrangian multipliers in sub-problems, facilitating the solution to the dual problem. However, the non-convex nature of the primal problem often renders the solution infeasible. Therefore, an approach for obtaining a feasible optimal or near-optimal solution becomes imperative.

Our primary objective is to maximize anticipated profits derived from the efficient operation of wind and hydro resources in electricity trading scenarios. The problem model, illustrated in Fig. 1, delineates the optimization challenge wherein the wind farm, coupled with a water storage system (wind/hydro), seeks to maximize profits by optimizing the active power cost delivered to the distribution network and the cost of pumping operations.

The most common form of energy storage is the accumulation of water pumping. Generally hydroelectric storage charges at night, meets peak demand for 4–8 h. Discharge varies, lasting hours to days, ensuring adaptability. Pumped storage is traditionally used for the management of energy and the development of spinning reserve, but also has applications in regulatory functions of frequency and operating

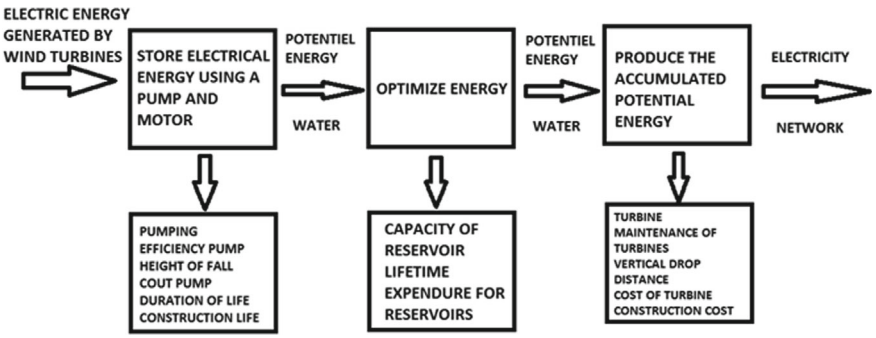


Fig. 1 Design a hydraulic storage system model

under partial load. The energy generated by the system is directed into the grid. The instantaneous total active power is characterized by the summation of hydropower (Ph) recovered at the turbine. Additionally, a substantial proportion of wind generation network output (Pw) contributes directly to the applied power at the pumping station and engages in power exchange (Pe) with the network during the time interval 't'.

(Pp) employed for pumping water to the upper pond level.

$$P(t) = P_w(t) + P_h(t) - P_e(t) - P_p(t) \geq 0 \tag{1}$$

The power output from a wind generator exhibits inherent variability owing to fluctuations in wind speed. Conversely, the station requires a stable power input. Given the power generated by the wind generator, denoted as Pw(t), and the target power to be supplied to the network, we establish a reference power as follows:

$$P_{ref}(t) = P(t) - P_w(t) \tag{2}$$

When the reference power is positive, indicating an excess of energy, the surplus must be stored. Conversely, if the reference power is negative, signifying an energy deficit, it needs to be compensated by utilizing stored energy.

The overall available power, denoted as P_v, is a combination of the power generated by the wind (Pw) and the power (Pp) employed for pumping water to the upper pond level within the time interval 't'.

$$P_v(t) = P_w(t) + P_p(t)$$

It appears that these first two constraints "(1)," and "(2)," can be formulated in a simple unique constraint than it is defined as follows:

$$-P_w(t) - P_h(t) + P_e(t) + P_p(t) + P_{ref}(t) \leq P(t)$$

The energy storage device is characterized by parameters such as power capacity, energy capacity, charging efficiency (η_p), and discharging efficiency (η_h). The constraints governing the temporal evolution of the available storage state in the tank can be succinctly expressed through two equations, considering the energy dynamics of the input and output ponds. The equation corresponding to the pumping operation is as follows:

$$E(t+1) = E(t) - \frac{P_h(t)}{\eta_h}\Delta t + \eta_p.P_p(t) \cdot \Delta t - s(t) \cdot \Delta t$$

And the other for the turbine operation:

$$E(t+1) = E(t) - \eta_p.P_p(t) \cdot \Delta t + \frac{P_h(t)}{\eta_h}\Delta t - s(t) \cdot \Delta t$$

where S(t) is the spillage discharge rate of the reservoir during time interval t.

Choosing not to operate the turbines and pumping simultaneously introduces a nuanced relationship between the available storage in the tank and the power flow in and out of the storage. This relationship is formally expressed as follows, delineating the intricate dynamics of the system.

$$E(t+1) = E(t) - \frac{P_h(t)}{\eta_h}\Delta t - s(t)\cdot\Delta t \,\text{For}\, \mathrm{P_p}(t) = 0$$

and

$$E(t+1) = E(t) - \eta_p.P_p(t)\cdot\Delta t - s(t)\cdot\Delta t \,\text{For}\, \mathrm{P_h}(t) = 0$$

To better express this constraint, we introduce a binary decision variable $\xi_{0,1}$.

With $\xi_{0,1} = \begin{cases} 1\, For\, \mathrm{P_p}(t) = 0 \\ 0\, For\, \mathrm{P_h}(t) = 0 \end{cases}$

than we have:

$$E(t+1) = E(t) - \xi_{0,1}\cdot\frac{P_h(t)}{\eta_h}\cdot\Delta t - \left(1-\xi_{0,1}\right).\eta_p.P_p(t)\cdot\Delta t - s(t)\cdot\Delta t$$

It guarantees the absence of parallel operation between turbines and pumping. The power consumption for water pumping is constrained within the limits of Pmp (upper limit) and PpM (lower limit) for the power station.

$$\mathrm{P^m{}_h} \le \mathrm{P_h}(t) \le \xi_{0,1}\mathrm{P^M{}_h} \text{ and } 0 \le \mathrm{P_p}(t) \le \mathrm{P^M{}_p}(1-\xi_{0,1})$$

The total active power delivered to the network, may have penalties, must at all times be higher than the minimum demand PL, and less than the maximum power exchange $\mathrm{P_{ex}}$

$$\mathrm{P_L} \le P(t) \le \mathrm{P_{ex}}$$

The constraint stipulates that the upper limit power for the hydroelectric generating unit is the lesser of its physical constraint (maximum turbine power, $\mathrm{P_hM}$) and the energy available in the reservoir.

$$\mathrm{P_h}(t) \le \min\left(\mathrm{P_hM}, \frac{\eta_h \mathrm{E}(t)}{\Delta t}\right)$$

The stored energy level within the reservoir must fall within the allowable limits stipulated for the reservoir.

$$0 \le E(t) \le \mathrm{E_m}$$

4 Objective Function

The primary objective is twofold: firstly, to minimize the overall generation cost while meeting system demand, spinning reserve requirements, and unit-specific constraints. Secondly, it seeks to maximize profits for the coupled wind farm and hydraulic storage system. Operating on a temporal unit of one hour, the planning horizon spans from one to ten days. Essentially, the aim is to optimize by maximizing the difference between the cost of active power supplied to the distribution network and the cost of pumping operations within the time interval 't'.

$$\sum_{t}(C \cdot P(t) - C_p.P_p(t))$$

5 Lagrangian Relaxation in the Realm of Hydraulic Storage Systems

5.1 Linear Programming and Lagrangian Relaxation

The term "linear integer programming" characterizes a subset of combinatorial optimization problems with integer variables. In this realm, the objective function assumes a linear structure, and the constraints are defined by linear inequalities [12].

The objective of the so-called Lagrangian relaxation methods is to give a tight evaluation of the subset S′ of treaty set S. The Lagrangian relaxation methods reduce to a considerable extent the number of nodes of the tree that it is necessary to visit. Spite of the extra calculate at each evaluation, the experience has shown that this approach led to relatively very efficient procedure.

The Linear Integer Programming (LIP) optimization problem written as:

$$(P)\begin{cases} \underset{x\geq 0}{\text{minimize}}\, cx \\ Ax \leq b \\ Bx \leq d \\ x\ integer \end{cases}$$

The expression, where vectors b, c, and d, along with matrices A and B of compatible dimensions, highlights the distinction between two types of constraints. Notably, the second type, $Bx \leq d$, is assumed to possess a unique and specialized structure [11].

While the Lagrangian relaxation is applied when we recognize in the matrix of constraints the difficult constraints. This method is to relax some constraints considered complicated. The relaxed constraints are reinjected into the objective function,

and weighted by coefficients named the Lagrange multipliers. The relaxation makes problems easier to solve.

Z * is the optimal solution of P. We assume that the resolution of the problem with only the constraints Ax ≤ b is easy to solve and that the introduction of constraints Bx ≤ d makes the problem more difficult. The idea of Lagrangian relaxation is to solve the problem using only the "easy" constraints. For this, we brought the "hard" constraints in the objective function by weighting a multiplier vector λ (Lagrangian multiplier). We define the Lagrangian relaxation of (P) relative to Ax ≤ b and a vector λ that is both nonnegative and conformable nonnegative vector λ as:

$$P(\lambda)\begin{cases} L(\lambda) = min\, Cx + \lambda(Bx - d) \\ Ax \le b \\ x\, integer \end{cases}$$

5.2 Formulation of the Problem with Lagrangian Relaxation

The fundamental concept behind the Lagrangian relaxation technique is to alleviate the system-wide constraints on demand and spinning reserve by employing Lagrange multipliers, resulting in the formulation of a two-level structure. The Lagrangian is then structured based on the cost function and constraints, manifesting as follows:

$$L = \sum_{t=1}^{T} \left\{ \begin{array}{l} \big(C.P(t) - C_p.P_p(t)\big) - \lambda(t)(P(t) - P_e(t) - P_p(t) \\ -P_{ref}(t) + P_w(t) + P_h(t)) \\ -\mu(t)\Big(-\xi_{0,1}.\dfrac{P_h(t)}{\eta_h}.\Delta t - \big(1-\xi_{0,1}\big).\eta_p.P_p(t).\Delta t - s(t).\Delta t - E(t+1) + E(t)\Big) \end{array} \right\}$$

Here, λ(t) and μ(t) represent Lagrangian multipliers linked to the energy produced by the system and the energy storage requisites at time 't', respectively. We define

$$\lambda(t) \equiv [\lambda(1), \lambda(2), \ldots\ldots\ldots\ldots, \lambda(T)]^t$$

$$\mu(t) \equiv [\mu(1), \mu(2), \ldots\ldots\ldots\ldots, \mu(T)]^t$$

Applying the duality theorem and leveraging the decomposable structure of (L), a two-level optimization problem can be established through a max–min formulation. Given multipliers λ and μ, the lower level encompasses individual subproblems related to pumping, hydro, and demand:

Pumping subproblem:

$$\min L_p, with\, L_p \equiv -C_p.P_p(t)) - \lambda(t)P(t) + \mu(t)\big(1-\xi_{0,1}\big).\eta_p.P_p(t)\cdot\Delta t$$

Hydro subproblem:

$$\min L_h, with\, L_h \equiv -\lambda(t)P_h(t)) + \mu(t)\xi_{0,1} \cdot \frac{P_h(t)}{\eta_h} \cdot \Delta t$$

Demand subproblem:

$$\min L_d, with\, L_d \equiv C.P(t) - \lambda(t)P(t) - \mu(t)(E(t) - E(t+1)$$

Allow $L_p{}^{*}(\lambda, \mu)$, $L_h{}^{*}(\lambda, \mu)$ and $L_d{}^{*}(\lambda, \mu)$ to signify, respectively, the optimal Lagrangian for the pumping subproblem, hydro subproblem, and demand subproblem, given specific values of λ and μ. Subsequently, the high-level dual problem can be expressed as:

$\max \Phi(\lambda, \mu)$, with $\Phi(\lambda, \mu) \equiv \sum_{t=1}^{T} \{L_p^{*}(\lambda, \mu) + L_h^{*}(\lambda, \mu) + L_d^{*}(\lambda, \mu) + \lambda(t)(P_e(t) - P_{ref}(t) - P_w(t)) - \mu(t).s(t) \cdot \Delta t\}$.

Subject to

$$\mu(t) \geq 0, t = 1,2, \ldots, T$$

To approach a near-optimal solution, effective algorithms are essential for tackling three subproblems, solving the dual problem and constructing a feasible solution.

6 Conclusion and Perspective

Our focus centered on formulating the management problem as a linear mixed-variable program in our contribution, we studied the constraints which it affects the problem in its particularity as storage and power generation systems but not in a way generally which had been treated by several studies in recent years. The general idea of our product is treating our linear programming problem with the Lagrangian relaxation method.

This study proposes a comprehensive solution to the challenges posed by the intermittent nature of wind energy. One key approach involves the formulation of a linear mixed-variable program, integrating advanced techniques such as Mixed Linear Programming (MLP) and Lagrangian relaxation. By utilizing Lagrangian relaxation, the study aims to optimize the operation of wind and hydro resources, balancing active power cost, pumping operations, and grid exchange. This involves solving subproblems related to pumping, hydro, and demand, employing Lagrange multipliers to address constraints efficiently.

To validate these approaches, in response to the challenges posed by the intermittent nature of wind energy, this article proposes to use a comprehensive approach, leveraging Artificial Intelligence (AI) to enhance the optimization strategies outlined

in this paper. The integration of advanced machine learning techniques, including recurrent neural networks and ensemble models, enables more accurate predictive modeling by analyzing historical wind patterns and weather data. Furthermore, the implementation of reinforcement learning algorithms facilitates dynamic, real-time optimization, allowing the system to adapt to current conditions for efficient energy production, storage, and distribution. Swarm intelligence, specifically particle swarm optimization and ant colony optimization, contributes to the optimization process by efficiently tuning Lagrangian relaxation parameters. Emphasizing transparency and interpretability, the incorporation of explainable AI models ensures that stakeholders can comprehend and trust the decision-making processes. Moreover, the utilization of cloud-based AI solutions addresses scalability concerns, making these advanced optimization techniques accessible for widespread adoption. This comprehensive synergy of predictive modeling, dynamic optimization, transparent decision-making, and scalability positions AI as a transformative force in advancing the efficiency and sustainability of wind energy systems. This synergy fosters a promising trajectory for the integration of AI in renewable energy practices.

References

1. wang, L., Chen, Q.: Challenges and opportunities in wind energy development: a planning perspective. Int. J. Renew. Energy Plann. Manag. **8**(2), 45–58 (2022)
2. Lu, S., Chowdhury, S.: Advancements in energy storage systems for renewable integration. Energies **13**(15), 3911 (2020)
3. Johnson, A., Smith, B.: Holistic approaches to energy storage in future renewable energy systems. J. Sustain. Energy **12**(4), 189–204 (2023)
4. Li, H., Wang, Z.: Optimization strategies for energy consumer and production systems: a comprehensive review. Int. J. Energy Optim. Eng. **5**(2), 78–91 (2020)
5. Patel, V., Gupta, N.: Large-scale hydro storage: a reliable buffer for intermittent renewable power sources. Renew. Energy Adv. **15**(3), 102–115 (2021)
6. El Ghmary, M., et al.: Offloading decisions in a mobile edge computing node with time and energy constraints. Int. J. Commun. Netw. Inf. Secur. **12**(1), 101–107 (2020)
7. Patel, V., Gupta, N.: Feasible solutions in non-convex optimization: a comprehensive review. Optim. Lett. **18**(3), 126–139 (2020)
8. Zhang, H., Li, X.: Advancements in lagrangian techniques for non-convex optimization problems. J. Optim. **15**(4), 210–225 (2022)
9. Wang, Q., Chen, Z.: Multiplier updating methods for improved Lagrangian relaxation in energy systems optimization. Int. J. Energy Optim. Eng. **9**(1), 32–47 (2021)
10. Smith, J., Johnson, M.: Profit maximization in coupled wind and hydro energy systems: a lagrangian approach. J. Renew. Energy Optim. **14**(2), 65–78 (2023)
11. Wang, Q., Li, X.: Structural insights into constrained optimization: unraveling special constraints in matrix formulations. J. Math. Optim. **18**(2), 75–88 (2023)
12. Chen, W., Liu, Y.: Linear integer programming: unveiling optimality in combinatorial constrained optimization. J. Optim. Adv. **15**(3), 120–135 (2022)
13. El Ghmary, M. et al.: An energy and latency trade-off for resources allocation in a MEC system. Int. J. Interact. Mobile Technol. **17**(20) (2023)

A Survey About Learning-Based Variable Speed Limit Control Strategies: RL, DRL and MARL

Asmae Rhanizar and Zineb El Akkaoui

Abstract In the domain of road traffic control, Variable Speed Limit strategies play a crucial role, addressing objectives like mobility enhancement, safety improvement, and environmental considerations. This survey extensively explores the landscape of Learning-based VSL control strategies, investigating the integration of Reinforcement Learning, Deep Reinforcement Learning and Multi-agent Reinforcement Learning techniques. Through a thorough examination of existing literature, we trace the evolution of these strategies, progressing from single-agent VSL approaches to intricate multi-agent VSL strategies. The survey systematically reviews studies that investigate the effectiveness of various Reinforcement Learning algorithms in VSL systems, providing insights into the challenges, advancements, and future directions of both single-agent and multi-agent VSL control strategies. In conclusion, this paper offers a critical review, highlighting key issues identified from the literature review, and suggests directions for future research that should be addressed in the next generation of Learning-based VSL control strategies.

1 Introduction

The establishment of speed limits, especially on highways, significantly influences both safe driving and efficient traffic flow. In this context, Variable Speed Limit (VSL) control strategies serve as proactive measures in traffic management, playing a crucial role in improving road safety and optimizing traffic flow [1].

Variable Speed Limit systems represent a dynamic approach to defining speed limits, considering factors including traffic conditions, road safety, weather, and the environment. Communication of speed limits to drivers is facilitated through Variable Message Signs (VMS). In the context of mixed traffic involving human-

A. Rhanizar (✉) · Z. El Akkaoui
National Institute of Posts and telecommunications, Rabat, Morocco
e-mail: rhanizar@inpt.ac.ma

Z. El Akkaoui
e-mail: elakkaoui@inpt.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_45

driven, connected, and autonomous vehicles, speed limits are conveyed either directly to drivers through connected vehicles' interfaces or autonomously applied in the case of self-driving vehicles. Throughout this survey, the notation VSL will be used to denote Variable Speed Limit systems.

Diverse VSL control strategies have been developed, spanning from initial rule-based reactive approaches to advanced proactive cooperative strategies. The latter exhibit the capability to anticipate the intricate and dynamic behavior of road traffic. Machine learning (ML), particularly Reinforcement Learning (RL) techniques, has emerged as a powerful tool in providing intelligent solutions for dynamic speed selection, offering promising solutions for modeling VSL agents [2].

In the landscape of road traffic control, the research community has made significant strides in developing and advancing VSL systems to address the complex dynamics of traffic flow and enhance road safety. However, as the field continues to evolve, there is a growing need to comprehensively survey the existing literature, methodologies, and advancements in Learning-based VSL control strategies, addressing both single-agent and multi-agent environments.

Unlike existing studies, such as the survey of Khondaker and Kattan [3] providing an overview of VSL systems, or the comprehensive study by KuÅ¡ic and al. [2] that offers a survey on the state of the art of RL applied to VSL, this study offers an in-depth exploration of the landscape of Learning-based VSL systems with a categorization of these systems into single-agent or multi-agent VSL control strategies, with a specific focus on the integration of RL, Deep Reinforcement Learning (DRL), and Multi-Agent Reinforcement Learning (MARL) techniques. By systematically examining the evolution of these strategies, their applications, and their impact on road safety and traffic efficiency, this review provides valuable perspectives on the present status of the field, identifies obstacles, showcases recent developments, and outlines potential avenues for future research. Consequently, it proves to be a valuable asset for researchers, practitioners, and policymakers involved in the intersection of reinforcement learning and traffic control.

The research utilizes a systematic literature review methodology, employing a keyword-based search to systematically locate primary studies within the search results. The review focuses on identifying and summarizing the RL methods and algorithms (RL, DRL or MARL) utilized to address VSL control strategies. The objectives of the identified strategies, including improving traffic efficiency or road safety, were identified for comprehensive analysis.

The remainder of this paper is organized as follows. Section 2 reviews VSL control strategies, exploring their diverse approaches and examining the advantages of implementing VSL in real-world control strategies. Section 3 introduces Learning-based VSL control strategies, investigating both single-agent and multi-agent setups, and presenting various techniques such as RL, DRL, and MARL for VSL systems. Section 4 initiates a discussion, emphasizing open issues that warrant exploration in future research.

2 VSL Overview

2.1 VSL Control Benefits

Speed on the roads directly influences traffic flow and plays a crucial role in maintaining efficient and safe traffic conditions. Indeed, higher speeds generally allow for smoother traffic flow and faster movement of vehicles through the road network. Appropriate speeds help minimizing congestion and slowdowns, thereby improving the overall efficiency of the transportation system. However, choosing moderate speeds reduces the risk of accidents and mitigates the severity of collisions in case of incidents. Thus, defining speed limits for a road network represents a challenge requiring a delicate balance between facilitating higher speeds to avoid congestion and regulating speeds to ensure safety. In response to this challenge, Variable Speed Limit control strategies have been extensively studied, demonstrating their impact in various works on improving mobility, safety, and environmental considerations.

- A study conducted in 2004 demonstrated the benefits of VSL in terms of road safety using a real-time accident prediction model integrated into a microscopic traffic simulation model, PARAMICS. The study showed that the deployment of VSL can reduce the accident risk by 5% to 17% based on simulating traffic conditions on a road section in Toronto, Canada [4].
- Another study assessing the impacts of VSL system on road safety demonstrated a decrease in total crash count of 32.23%, with a standard deviation of 3.58%, on Interstate 5 in Seattle, United States. The study conducted a comprehensive before-and-after Bayesian analysis based on 9,787 accidents over a 72-month study period [5].
- A research introduced a VSL system (FC-VSL) designed to consider fuel consumption, with the goal of minimizing the average fuel usage of vehicles on highways in actual traffic scenarios. Simulation results showcased that the FC-VSL has the capability to substantially decrease the average fuel consumption of vehicles, approaching outcomes close to the optimal levels [6]. This study showed that VSL systems can be deployed as a solution to minimize the carbon footprint of vehicles.
- Zhibin and al. introduced a VSL control strategy designed to dynamically adapt speed limits according to prevailing traffic and weather conditions. The research showcased a decrease in the risk of secondary collisions across different weather scenarios. The time exposed time-to-collision (TET) experienced a reduction ranging from 41.45% to 50.74%, while the time integrated time-to-collision (TIT) saw a decrease between 38.19% and 41.19% [7].
- Within the realm of connected vehicles, Yang and al. introduced a driving simulator testbed to evaluate the influence of implementing a Connected Vehicle VSL (CV-VSL) system in Wyoming, United States, on the behavior of truck drivers during adverse weather conditions. The simulation outcomes revealed that, particularly when advisory speed limits were set below 88.5 km/h, participants typically adhered to the VSL presented on the Connected Vehicle's Human-Machine Interface [8].

2.2 VSL Control Strategies

Numerous VSL control strategies have been proposed, spanning from initial rule-based reactive methodologies to the most sophisticated proactive cooperative strategies. The latter approaches inherently have the ability to anticipate the complex and dynamic behavior of road traffic. Machine learning, specifically RL techniques, have been widely applied to provide intelligent solutions for dynamic speed selection, offering promising alternatives for modeling VSL controllers. VSL control strategies can be broadly categorized into two main types: Reactive Rule-based VSL and Proactive VSL.

- **Reactive Rule-Based VSL**: This represents a basic fundamental method for implementing VSL systems, primarily focusing on harmonizing speed differences and stabilizing traffic flow. Such VSL implementation involves a control system governed by pre-established rules, where control strategies are predetermined based on specific scenarios. For example, a rule could dictate a reduction in the speed limit during adverse weather conditions or traffic congestion. Rule formulation frequently relies on human expertise to respond to traffic or weather conditions. The limitation of these strategies is their incapacity to dynamically adapt to varying traffic conditions, particularly in intricate and dynamic traffic environments [9–11].
- **Proactive VSL**: These encompass advanced VSL control strategies designed to overcome the constraints of rule-based VSL. In these advanced approaches, predictions of future traffic conditions are made to anticipate potential disruptions. Corrective VSL strategies are subsequently incorporated into the system to proactively mitigate congestion, diverting traffic away from potential bottlenecks and resolving shockwaves before they propagate to disrupt traffic, ensuring a safer traffic environment. VSL strategies can also be applied cooperatively, as any control action may influence the flow in an adjacent section of the network. Conversely, proactive VSL strategies are typically data-driven, utilizing real-time data for anticipatory and dynamic speed limit decisions. Artificial Intelligence enhances Learning-based VSL control strategies with advanced capabilities in data analysis, prediction, and decision-making, thereby strengthening the overall effectiveness of these systems. In response, several studies have proposed different control strategies based on machine learning algorithms, particularly RL [3].

3 Learning-Based VSL Control Strategy

3.1 Learning-Based Single-Agent VSL

In the realm of traffic control, the integration of Learning-based approaches has emerged as a transformative paradigm. This shift towards intelligent systems is partic-

ularly evident in the context of VSL control strategies. Unlike traditional rule-based methods, Learning-based strategies harness the power of Artificial Intelligence to dynamically adjust speed limits based on real-time conditions and evolving traffic scenarios. In this sub-section, we focus on the studies that addressed Learning-based single-agent VSL systems to enhance the adaptability and responsiveness of traffic control policies. These approaches employ Reinforcement Learning and its extension, Deep Reinforcement Learning, to empower the VSL agent with the ability to learn optimal speed adjustments through interactions with their environment [12, 14, 15]. By delving into the principles of RL and DRL, we aim to unravel the potential of this intelligent approach in achieving not only the conventional goals of safety and mobility but also in addressing the complex trade-offs inherent in real-world traffic control scenarios.

3.1.1 RL Foundations and Notations

Reinforcement Learning is a subset of machine learning in which an agent takes actions within an environment to achieve specific goals. The agent acquires knowledge from its experiences via trial and error, where it receives rewards or punishments depending on the effectiveness of its actions [16]. The evaluation of action quality considers not only the immediate rewards but also the potential delayed rewards they may generate. RL is formulated in the mathematical representation of a decision-making process called "Markov Decision Process" (MDP) [17], defined by:

- Time t
- State s_t belonging to the state space S ($s_t \in S$)
- A transition function $T(s_t, a_t, s_{t+1})$, which is the probability that an action a_t causes the transition to state s_{t+1} from state s_t, i.e., $P(s_{t+1}|s_t, a_t)$.
- A reward function $R(s_t, a_t, s_{t+1})$.

In RL, the agent aims to learn an optimal policy denoted as $\pi^* : S \longrightarrow A$ where π represents the policy. The agent aims to maximize the anticipated reward, prompting adjustments to the policy to reach the optimal configuration that fulfills this objective (Fig. 1).

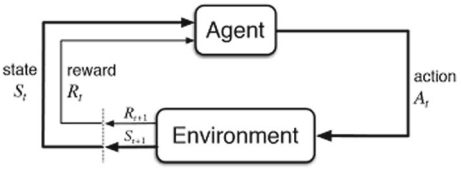


Fig. 1 RL notations: At each time step t, a RL agent observes a state s_t in the state space S, from which it chooses an action a_t in the action space A, following a policy π. The agent receives a reward r_t when transitioning to state s_{t+1} according to the dynamics of the environment

Q-Learning

Q-learning [18] is a reinforcement learning technique used to solve problems where an agent must make sequential decisions in an uncertain environment. As in RL, the transition and reward functions are unfamiliar since the agent lacks prior knowledge of the model (model-free), RL agents learn Q-values instead of state values. A Q-value is a function of the "state-action" pair that returns a real value: $Q : S \times A \longrightarrow R$.

It has been demonstrated by Watkins and Dayan [18] that Q-learning converges to the optimal policy with a probability of 1 as long as all actions are repeatedly sampled in all states and Q-values are discretely represented. This underscores the importance of exploration to ensure that all actions in all states are sampled. Consequently, the exploration-exploitation trade-off dilemma emerges, where an agent must prioritize actions that, based on its past experiences, have yielded the highest rewards. However, to discover these actions, the agent must engage in new actions (exploration) that it has not previously attempted. Eventually, the agent must leverage its acquired knowledge (exploitation) to maximize rewards. A common strategy for exploration is to randomly select actions with a small probability, known as "ϵ-greedy" exploration [19].

3.1.2 RL for Single-Agent VSL Systems

RL has been applied to a wide range of complex problems that cannot be solved with other machine learning algorithms, especially in the fields of robotics, games, Intelligent Transportation Systems (ITS), and other domains. Notably, RL has been extensively explored in the literature as a solution for VSL control strategies. KuÅ¡iÄ‡ and al. conducted a comprehensive literature review on the application of RL to VSL, summarizing recent works in their paper [2]. The following are examples of the proposed Learning-based single-agent VSL systems based on RL, designed to meet safety, mobility, or environmental requirements:

- A study conducted by Zhibin and al. proposed a VSL control algorithm based on RL to reduce the risks of accidents related to oscillations. The results showed that after the learning process, RL-based VSL regulation successfully reduced accident risks by 19.4% [12].
- Another approach has been presented in [14], suggesting a RL-based control approach designed to enhance traffic efficiency using VSL against congestion on the highway with a mobility-oriented goal. The proposed VSL control approach is tested using a macroscopic traffic simulation model METANET to represent the real dynamics of traffic flow.
- In a mixed traffic scenario, a research paper introduced a VSL system based on Q-learning with Connected and Autonomous Vehicles acting as mobile sensors. This approach, combined with speed transition matrices for state estimation, surpassed other control strategies and enhanced key macroscopic traffic metrics such as Total Time Spent and Mean Travel Time [15].

Recent advancements have significantly expanded the frontiers of RL through the integration of deep neural networks, leading us into the domain of Deep Reinforcement Learning. While traditional RL methods demonstrate proficiency in specific scenarios, the integration of deep learning techniques empowers agents to navigate more intricate and high-dimensional environments.

3.1.3 Deep Reinforcement Learning

In recent years, research in the field of Deep Learning has demonstrated its promise and power as a tool for automatic feature extraction from raw data, such as the raw pixels of an image [20–22]. DL has accelerated progress in RL, with the incorporation of deep learning algorithms into RL defining the field of Deep Reinforcement Learning. DL enables RL to scale to decision-making problems that were previously considered intractable [23, 24].

In particular, the problem of dimensionality (known as the "Curse of Dimensionality"), arising from large state S and action A spaces, preventing the learning of Q-value estimates for each "state-action" pair independently, as in traditional tabular Q-Learning. Consequently, DRL employs deep neural networks to model the components of reinforcement learning, and the network parameters undergo training using gradient descent to minimize a relevant loss function. Within the realm of DRL, several algorithms have been developed, each tailored to address specific challenges and problem domains. Noteworthy algorithms that have significantly contributed to the field are highlighted here, and for a more in-depth exploration of fundamental concepts and a variety of DRL algorithms, comprehensive surveys can be found in: [25–28].

Deep-RL for single-agent VSL systems

The integration of DRL into VSL systems enables the development of adaptive and proactive control strategies that can respond in real-time to changing traffic conditions. Research in this domain explores various DRL architectures, including Deep Q Networks (DQN) [23], Policy Gradient Methods [29], and actor-critic models [30], to tailor them to the specific challenges of VSL systems. These applications have shown promising results in simulation studies and real-world scenarios, demonstrating the potential for DRL to revolutionize the way VSL systems operate and contribute to the overall improvement of traffic management strategies:

- Wu and al. [31] introduced a DRL model designed for implementing a differential variable speed limit control strategy, enabling dynamic and different speed limits for individual lanes. The authors introduced a unique actor-critic architecture to train the model in learning multiple discrete speed limits within a continuous action space. Evaluation of the DRL model proposed for DVSL control was conducted on a simulated freeway with recurrent bottlenecks. The outcomes revealed that the proposed controller effectively enhanced the safety, mobility, and environmental sustainability of the freeway studied.

- In their research, Ke and al. evaluated the effectiveness of a transfer learning algorithm to improve the adaptability of a VSL control system based on DRL. They introduced a Double Deep Q Network VSL control strategy aimed at improving traffic mobility. The transferred DDQN VSL strategy exhibited a notable decrease in the Total Time Spent (TTS), ranging from 26.02% to 67.37% [32].
- As part of an integrated control approach, Peng et al. introduced a unified system for secondary crash prevention by combining VSL and Lane Change Guidance system through distributed DRL [33]. The performance of the combined controller was assessed in terms of safety and mobility. The integrated controller's performance, evaluated in terms of safety and mobility, surpassed that of individual sub-controllers, highlighting its overall superiority.
- In a recent study, Chen and al. proposed a DRL-VSL control strategy to alleviate congestion and enhance safety [34]. The authors introduced a twin delayed deep deterministic policy gradient based VSL solution using the actor-critic model. The simulation outcomes indicate that employing TD3 model for VSL control was successful in decreasing average travel time and enhancing the throughput of passing vehicles.
- In a heterogeneous traffic setting, Lu and al. introduced a lane-level VSL control strategy utilizing an actor-critic architecture. The authors introduced a twin delayed deep deterministic policy gradient method to train the framework introducing a hybrid reward considering both traffic safety and efficiency. The controller proposed exhibits superior performance in terms of both traffic safety and mobility [35].

3.2 Learning-Based Multi-agent VSL

While VSL systems are usually implemented in specific road segments that are experiencing bottlenecks often utilizing a single VSL [12, 14], real-world scenarios, particularly those involving intricate bottleneck situations, may require deploying multiple control agents to effectively manage a broader road network and address global objectives such as safety, mobility or the environment. In this context, the increasing interest in optimizing transportation systems for efficiency, resource conservation, and ecological sustainability is addressed by Multi-Agent Systems (MAS). Numerous researchers have explored the implementation of MAS for Intelligent Transportation Systems covering various aspects such as traffic management [36, 37], traffic control [38, 39] and transportation simulation systems [40, 41].

While numerous studies have evaluated the advantages of VSL for a single road segment, real-world traffic scenarios often require deploying multiple VSL agents to manage expansive road networks. A strategy involving the coordination of multiple agents can address these challenges by assigning speed limits across various upstream sections of the motorway, facilitating smoother speed transitions. A recent trend in literature involves addressing multi-agent systems using Multi-Agent Reinforcement

Learning, providing a Learning-based approach to enhance the coordination and decision-making capabilities of these multiple VSL agents.

In a multi-agent setting, a road network serves as the environment for multiple VSL agents collaboratively working towards global objectives that encompass traffic flow, safety, and environmental considerations.

3.2.1 Multi-agent Reinforcement Learning

MARL introduction

In a data-driven approach, multi-agent systems have been recently addressed in literature using MARL, which extends traditional RL techniques to environments with multiple agents [42]. In this setup, each agent interacts with the environment, learning to make decisions based on its observations and experiences. MARL can be applied to scenarios with cooperative, competitive, or mixed dynamics among multiple agents. As traffic control scenarios often involve multiple interacting VSL agents, the exploration of the application of MARL in traffic control has garnered increasing attention in recent years [42, 43].

MARL variants

Various MARL variants have been proposed in the literature [42]. One early approach, is **independent learning**, treating each agent as an autonomous learner, where the actions of other agents are considered as part of the environment [44]. However, this introduces a challenge of non-stationarity due to the impact of other agents' actions on local interests and environment transitions.

An approach to tackle the non-stationarity issue involves using a **fully observable critic**, incorporating the observations and actions of all agents. This resolves the non-stationarity of the critic serving as a robust leader for local actors. In this context, Lowe and al. [45] propose Multi-Agent Deep Deterministic Policy Gradient (MADDPG) where each agent trains a deep deterministic policy gradient (DDPG) algorithm. The actor observes its local environment, while the critic has the ability to access the whole observations, actions, and the target policies of all agents during the training process. Various extensions of MADDPG algorithm were proposed, including attention MADDPG (ATT-MADDPG) [47], generative cooperative policy network MADDPG (MADDPG-GCPN) [46] and Recurrent MADDPG (R-MADDPG) [48].

In a different investigation, Foerster and al. [49] propose COMA, a framework that incorporates a singular centralized critic utilizing the global state, the aggregate vector of actions, and a collective reward. This common critic is utilized by all agents, and each actor undergoes localized training specific to its respective agent, utilizing the local observation-action history.

Another type of MARL variants is **Value Function Factorization**, which assumes that some agents may become uninvolved in learning and coordinating as intended, potentially causing system failure. Algorithms within this classification tackle this

issue by identifying the contribution of each agent to the collective reward and subsequently isolating its portion from it. The VDN method, introduced by Sunehag and al. [50], assesses the influence of each agent on the observed collective reward. QMIX shares the action-value function during centralized training and incorporates additional information from the global state [51].

Algorithms employing a centralized critic may encounter challenges related to dimensionality as the number of agents increases since they concatenate all local observations. To counter this issue, the **Consensus** approach was introduced, enabling agents to share information exclusively with a subset of agents (neighbors) in order to achieve consensus. Macua and al. [52] propose a completely distributed actor-critic algorithm known as Diff-DAC. In this method, agents communicate their value-policy parameters exclusively to neighboring agents, leading to a collective convergence on a shared policy. Zhang & al. suggest a fully decentralized MARL where each agent makes individual decisions based on local observations and messages received from neighbors [53]. Consensus updates via communication are implemented to achieve a decentralized critic.

In a different investigation, for environments allowing communications between agents, numerous MARL algorithms were proposed in the literature to allow agents to learn a communication policy to send information through the network. These algorithms take into account the learning duration for message transmission, the message type, and the destination agents. Foerster & al. propose Reinforced Inter-Agent Learning -RIAL- and Differentiable Inter-Agent Learning -DIAL-for acquiring communication protocols among multiple agents sensing and operating in environments [54]. Jorge & al. expand the DIAL algorithm by permitting communication of varying sizes, introducing incremental noise on communication channels to facilitate agents in learning a symbolic language, and restricting the sharing of parameters among agents [55]. They present the outcomes of their algorithm in the context of a modified version of the “Guess Who?” game.

In summarizing the diverse variants of MARL, each tailored to unique challenges and scenarios, the next step is to explore how these MARL approaches can be effectively applied to the domain of VSL control. Understanding the intricacies and nuances of MARL variants sets the stage for a deeper examination of their practical implementations and contributions to the optimization of VSL systems.

MARL for multi-agent VSL systems

While numerous studies have assessed the advantages of VSL for a single road segment, real-world traffic scenarios often involve the deployment of multiple VSL controllers to manage expansive road networks. A strategy involving the coordination of multiple agents can address these issues by assigning speed limits across various upstream sections of the motorway, facilitating more seamless speed transitions. Indeed, MARL has been explored by researchers for the development of diverse traffic control strategies [56–58]. Notably, MARL in the context of VSL traffic control has garnered attention in a limited number of studies (Fig. 2):

To the best of our knowledge, Wang & al. were pioneers in using MARL for VSL traffic control, introducing a multi-agent Vehicle-to-Infrastructure (V2I) VSL

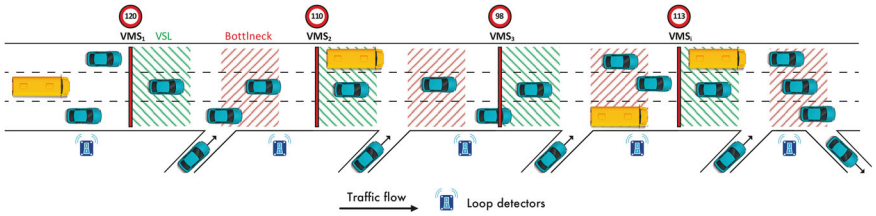


Fig. 2 Multi-agent VSL environment: illustrates a highway road network controlled by VSL units. The road network incorporates multiple ramps that may cause bottleneck situations in different sections. Speeds defined by the VSL agents are displayed on the Variable message Signs (VMS) screens. Loop detectors on roads provide agents with information about traffic dynamics

system based on a modified Deep Q-Learning (DQL) algorithm to optimize freeway traffic mobility and safety [59]. KuÅ¡iÄ‡ and al. proposed a collaborative distributed spatial-temporal multi-agent VSL (DWL-ST-VSL) employing a dynamically adjusting mechanism for VSL zones through a distributed W-learning algorithm [60]. Zheng and al. introduced a MARL-VSL Framework to alleviate congestion and enhance collaboration among VSL controllers. The researchers used the MADDPG algorithm developing a centralized training approach and decentralized execution for VSL control in different bottlenecks [61]. Zhang and al. suggested a Multi-Agent Reinforcement-learning for large-scale VSL (MARVEL), using the MAPPO algorithm (Multi-Agent Proximal Policy Optimization). The study's reward structure was designed to incorporate adaptability to traffic conditions, alongside considerations for safety and mobility [62]. Fang and al. proposed a MAPPO-based VSL control for the Motorway-Urban Merging Bottlenecks to streamline traffic flow and reduce emissions [63].

4 Discussion

The survey of learning-based VSL, which includes both single-agent and multi-agent paradigms (Reinforcement Learning, Deep Reinforcement Learning, Multi-Agent Reinforcement Learning) , unveiled promising alternatives for intelligent speed selection. Table 1 chronologically lists the most representative approaches for VSL design based on RL, DRL, or MARL techniques, mentioning the algorithm used and the type of data employed for the evaluation (Real-Data Vs Simulation). It also demonstrates an increasing interest in the application of RL paradigms to VSL control strategies.

The completed survey has identified several potential research directions to address the limitations of existing methods. Reinforcement Learning has provided valuable insights into the dynamic nature of speed control, showcasing the capacity of single-agent approaches to optimize speed limits based on individual objectives. However, existing single-agent RL-VSL control frameworks face limitations, particularly in handling the balance between concurrent objectives, such as safety and

Table 1 VSL Learning-based Frameworks comparison

Paper	Year	Type	RL algorithm	Objective	Data
[15]	2023	RL VSL agent	Q-Learning	Mobility	SUMO simulator
[34]	2023	DRL VSL agent	TD3	Mobility	Simulation
[35]	2023	DRL VSL agent	TD3	Mobility, Safety	Simulation
[61]	2023	MARL VSL agent	MADDPG	Mobility	Cell Transmission Model (CTM)
[62]	2023	MARL VSL agent	Multi-Agent Proximal Policy Optimization (MAPPO)	Mobility, Safety	TransModeler simulator
[14]	2022	RL VSL agent	Q-Learning	Mobility	METANET simulator
[33]	2022	DRL VSL agent	Distributed DRL	Safety	Simulation
[60]	2021	MARL VSL agent	Distributed W-learning (DWL)	Mobility	SUMO simulator
[12]	2021	RL VSL agent	Q-Learning	Mobility	Cell Transmission Model (CTM)
[32]	2021	DRL VSL agent	DDQN	Mobility	Cell Transmission Model (CTM)
[31]	2020	DRL VSL agent	DDPG	Mobility	SUMO simulator
[59]	2019	MARL VSL agent	Deep Q-Learning	Mobility, Safety	Cell Transmission Model (CTM)

mobility. While most studies focus on a single objective VSL control strategy [12, 14, 15], achieving an efficient control strategy necessitates addressing the delicate equilibrium between safety and mobility, which poses a significant challenge.

Limited studies delve into the multi-objective single-agent VSL control strategy, especially the critical balance between traffic efficiency and road safety [35]. In particular, Reward engineering emerges as a crucial aspect to consider, allowing for the customization of multi-objective VSL agents. In a recent study, Lu and al. [35] designed a hybrid reward function that synchronously considers traffic safety and traffic efficiency in a bottleneck area within a connected automated vehicle highway (CAVH) environment. The findings demonstrated that the proposed method effec-

tively reduces crash risk and improves traffic efficiency simultaneously. However, further exploration of multi-objective single-agent VSL systems is still needed.

Regarding multi-agent VSL control strategies, few recent studies have applied MARL algorithms to explore their potential [56, 60–63], yet further investigation is required to explore the use of additional MARL variants for addressing the intricacies of complex traffic environments. On the other hand, scalability poses a challenge for extending VSL control strategies to real-world, complex multi-agent scenarios. The MARL-VSL control strategies based on the Centralized Learning and Decentralized Execution (CLDE) approach face limitations when scaling up the number of agents, primarily due to coordination and communication challenges among a large agent population [61, 62].

It is noteworthy that the evaluation of Learning-based VSL proposed strategies predominantly relies on simulations, encompassing both microscopic and macroscopic approaches. A notable exception is the work of Zhang et al. [62], who pioneered the testing of a learning-based VSL using real-world data. Acknowledging the scarcity of studies utilizing real data to test and evaluate VSL agents, we recognize the need to address this gap in future research.

5 Conclusion

This survey navigated through the diverse landscape of Variable Speed Limit, uncovering motivations, control strategies, and the evolving Learning-based approaches. From the early rule-based reactive methods to advanced proactive cooperative strategies, our exploration showcased the pivotal role of VSL in anticipating the complexities of road traffic environments. The examination of Learning-based VSL control strategies, encompassing single-agent and multi-agent perspectives using Reinforcement Learning techniques, revealed promising avenues for intelligent speed selection. However, as we look at the current state of the field, certain challenges and open issues come to light. As we peer into the future, the integration of Learning-based VSL control strategies into real-world traffic scenarios holds tremendous promise for improving road safety, mobility, and overall traffic management.

References

1. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition (2015)
2. Kušić, K., Ivanjko, E., Gregurić, M., Miletić, M.: An overview of reinforcement learning methods for variable speed limit control. Appl. Sci. **10**, 4917 (2020)
3. Khondaker, B., Kattan, L.: Variable speed limit: an overview. Transp. Lett. **7**(5), 264–278 (2015)
4. Lee, C., Hellinga, B., Saccomanno, F.: Evaluation of variable speed limits to improve traffic safety. Transp. Res. Part C Emerg. Technol. **14**, 213–228 (2006)

5. Pu, Z., Li, Z., Jiang, Y., Wang, Y.: Full Bayesian before-after analysis of safety effects of variable speed limit system. IEEE Trans. Intell. Transp. Syst. **22**, 964–976 (2021)
6. Liu, B., Ghosal, D., Chuah, C.-N., Zhang, H.M.: Reducing greenhouse effects via fuel consumption-aware variable speed limit (FC-VSL). IEEE Trans. Veh. Technol. **61**, 111–122 (2012)
7. Li, Z.: Development of a variable speed limit strategy to reduce secondary collision risks during inclement weathers. Accident Anal. Prevent. (2014)
8. Yang, G., Ahmed, M.M., Gaweesh, S.: Impact of variable speed limit in a connected vehicle environment on truck driver behavior under adverse weather conditions: driving simulator study. Transp. Res. Record J. Transp. Res. Board **2673**, 132–142 (2019)
9. Van Den Hoogen, E.: Control by variable speed signs: results of the Dutch experiment. In: Seventh International Conference on 'Road Traffic Monitoring and Control', vol. 1994, (London, UK), pp. 145–149. IEE (1994)
10. Zackor, H.: Speed limitation on freeways: traffic-responsive strategies. In: Concise Encyclopedia of Traffic & Transportation Systems, pp. 507–511, Elsevier (1991)
11. Piao, J., McDonald, M.: Safety impacts of variable speed limits - a simulation study. In: 2008 11th International IEEE Conference on Intelligent Transportation Systems, (Beijing, China), pp. 833–837, IEEE (2008)
12. Li, Z., Xu, C., Guo, Y., Liu, P., Pu, Z.: Reinforcement learning-based variable speed limits control to reduce crash risks near traffic oscillations on freeways. IEEE Intell. Transp. Syst. Mag. **13**(4), 64–70 (2021)
13. Zhu, F., Ukkusuri, S.V.: Accounting for dynamic speed limit control in a stochastic traffic environment: a reinforcement learning approach. Transp. Res. Part C Emerg. Technol. **41**, 30–47 (2014)
14. Han, Y., Hegyi, A., Zhang, L., He, Z., Chung, E., Liu, P.: A new reinforcement learning-based variable speed limit control approach to improve traffic efficiency against freeway jam waves. Transpo. Res. Part C: Emerg. Technol. **144**, 103900 (2022)
15. Vrbanić, F., Tišljarić, L., Majstorović, Ž, Ivanjko, E.: Reinforcement Learning-based dynamic zone placement variable speed limit control for mixed traffic flows using speed transition matrices for state estimation. Machines **11**, 479 (2023)
16. Kaelbling, L.P., Littman, M.L., Moore, A.W.: Reinforcement learning: a survey. J. Artif. Intell. Res. **4**, 237–285 (1996)
17. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming, 1st edn. Wiley Series in Probability and Statistics, Wiley (1994)
18. Watkins, C.J.C.H., Dayan, P.: Q-learning. Mach. Learn. **8**, 279–292 (1992)
19. Williams, R.J., Baird, L.C.: Tight performance bounds on greedy policies based on imperfect value functions (1993)
20. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Commun. ACM **60**(6), 84–90 (2017)
21. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**, 436–444 (2015)
22. Shinde, P.P., Shah, S.: A review of machine learning and deep learning applications. In: 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), pp. 1–6 (2018)
23. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. Nature **518**, 529–533 (2015)
24. Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., Hassabis, D.: Mastering the game of Go with deep neural networks and tree search. Nature **529**, 484–489 (2016)
25. Li, Y.: Deep Reinforcement Learning: An Overview (2017)

26. Arulkumaran, K., Deisenroth, M.P., Brundage, M., Bharath, A.A.: Deep reinforcement learning: a brief survey. IEEE Signal Process. Mag. **34**(6), 26–38 (2017)
27. Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., Meger, D.: Deep reinforcement learning that matters. Proceedings of the AAAI Conference on Artificial Intelligence, vol. 32, no. 1 (2018)
28. Mousavi, S.S., Schukat, M., Howley, E.: Deep reinforcement learning: an overview. In: Bi, Y., Kapoor, S., Bhatia, R. (eds.) Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016, pp. 426–440. Springer International Publishing, Cham (2018)
29. Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., Wierstra, D.: Continuous control with deep reinforcement learning (2019). arXiv:1509.02971 [cs, stat]
30. Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T.P., Harley, T., Silver, D., Kavukcuoglu, K.: Asynchronous Methods for Deep Reinforcement Learning (2016). Publisher: arXiv Version Number: 2
31. Wu, Y., Tan, H., Qin, L., Ran, B.: Differential variable speed limits control for freeway recurrent bottlenecks via deep actor-critic algorithm. Transp. Res. Part C: Emerg. Technolo. **117**, 102649 (2020)
32. Ke, Z., Li, Z., Cao, Z., Liu, P.: Enhancing transferability of deep reinforcement learning-based variable speed limit control using transfer learning. IEEE Trans. Intell. Transp. Syst. **22**(7), 4684–4695 (2021)
33. Peng, C., Xu, C.: Combined variable speed limit and lane change guidance for secondary crash prevention using distributed deep reinforcement learning. J. Transp. Safety Secur. **14**(12), 2166–2191 (2022)
34. Chen, X., Jiang, J., Yang, J., Liu, Y.: Deep reinforcement learning based lane-level variable speed limit control. In: 2023 9th International Conference on Control Science and Systems Engineering (ICCSSE), pp. 98–104 (2023)
35. Lu, W., Yi, Z., Gu, Y., Rui, Y., Ran, B.: TD3LVSL: A lane-level variable speed limit approach based on twin delayed deep deterministic policy gradient in a connected automated vehicle environment. Transp. Res. Part C: Emerg. Technol. **153**, 104221 (2023)
36. Adler, J.L., Satapathy, G., Manikonda, V., Bowles, B., Blue, V.J.: A multi-agent approach to cooperative traffic management and route guidance. Transp. Res. Part B: Methodol. **39**, 297–318 (2005)
37. Hamidi, H., Kamankesh, A.: An approach to intelligent traffic management system using a multi-agent system. Int. J. Intell. Transp. Syst. Res. **16**, 112–124 (2018)
38. Hirankitti, V., Krohkaew, J., Hogger, C.: A Multi-Agent Approach for Intelligent Traffic-Light Control (2007)
39. Ikidid, A., Abdelaziz, E.F., Sadgal, M.: Multi-agent and fuzzy inference-based framework for traffic light optimization. Int. J. Interact. Multimedia Artif. Intell. **8**(2), 88 (2023)
40. Cetin, N., Nagel, K., Raney, B., Voellmy, A.: Large-scale multi-agent transportation simulations. Comput. Phys. Commun. **147**, 559–564 (2002)
41. Tao, C., Huang, S.: An extensible multi-agent based traffic simulation system. In:2009 International Conference on Measuring Technology and Mechatronics Automation, (Zhangjiajie, Hunan, China), pp. 713–716, IEEE (2009)
42. OroojlooyJadid, A., Hajinezhad, D.: A Review of Cooperative Multi-Agent Deep Reinforcement Learning (2021)
43. Nguyen, T.T., Nguyen, N.D., Nahavandi, S.: Deep reinforcement learning for multiagent systems: a review of challenges, solutions, and applications. IEEE Trans. Cybern. **50**, 3826–3839 (2020)
44. Kilinc, O., Montana, G.: Multi-agent Deep Reinforcement Learning with Extremely Noisy Observations (2018)
45. Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., Mordatch, I.: Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments (2020)
46. Ryu, H., Shin, H., Park, J.: Multi-Agent Actor-Critic with Generative Cooperative Policy Network (2018)

47. Mao, H., Zhang, Z., Xiao, Z., Gong, Z.: Modelling the Dynamic Joint Policy of Teammates with Attention Multi-agent DDPG (2018). Publisher: arXiv Version Number: 1
48. Wang, R.E., Everett, M., How, J.P.: R-MADDPG for Partially Observable Environments and Limited Communication (2020). Publisher: arXiv Version Number: 2
49. Foerster, J., Farquhar, G., Afouras, T., Nardelli, N., Whiteson, S.: Counterfactual Multi-agent Policy Gradients (2017)
50. Sunehag, P., Lever, G., Gruslys, A., Czarnecki, W.M., Zambaldi, V., Jaderberg, M., Lanctot, M., Sonnerat, N., Leibo, J.Z., Tuyls, K., Graepel, T.: Value-Decomposition Networks For Cooperative Multi-Agent Learning (2017)
51. Rashid, T., Samvelyan, M., de Witt, C.S., Farquhar, G., Foerster, J., Whiteson, S.: QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning (2018). Publisher: arXiv Version Number: 2
52. Macua, S.V., Tukiainen, A., Hernández, D.G.-O., Baldazo, D., de Cote, E.M., Zazo, S.: Diff-DAC: Distributed Actor-Critic for Average Multitask Deep Reinforcement Learning (2017)
53. Zhang, K., Yang, Z., Liu, H., Zhang, T., Başar, T.: Fully Decentralized Multi-agent Reinforcement Learning with Networked Agents (2018)
54. Foerster, J.N., Assael, Y.M., de Freitas, N., Whiteson, S.: Learning to Communicate with Deep Multi-agent Reinforcement Learning (2016)
55. Jorge, E., Kågebäck, M., Johansson, F.D., Gustavsson, E.: Learning to Play Guess Who? and Inventing a Grounded Language as a Consequence (2016)
56. Le, N.-T.-T.: Multi-agent reinforcement learning for traffic congestion on one-way multi-lane highways. J. Inf. Telecommun. **7**, 255–269 (2023)
57. Calvo, J.A., Dusparic, I.: Heterogeneous Multi-agent Deep Reinforcement Learning for Traffic Lights Control
58. Wang, X., Ke, L., Qiao, Z., Chai, X.: Large-scale traffic signal control using a novel multiagent reinforcement learning. IEEE Trans. Cybern. **51**, 174–187 (2021)
59. Wang, C., Zhang, J., Xu, L., Li, L., Ran, B.: A new solution for freeway congestion: cooperative speed limit control using distributed reinforcement learning. IEEE Access **7**, 41947–41957 (2019)
60. Kušić, K., Ivanjko, E., Vrbanić, F., Gregurić, M., Dusparic, I.: Spatial-temporal traffic flow control on motorways using distributed multi-agent reinforcement learning. Mathematics **9**, 3081 (2021)
61. Zheng, S., Li, M., Ke, Z., Li, Z.: Coordinated variable speed limit control for consecutive bottlenecks on freeways using multiagent reinforcement learning. J. Adv. Transp. **2023**, 1–19 (2023)
62. Zhang, Y., Quinones-Grueiro, M., Zhang, Z., Wang, Y., Barbour, W., Biswas, G., Work, D.: MARVEL: Multi-agent Reinforcement-Learning for Large-Scale Variable Speed Limits (2023)
63. Fang, X., Péter, T., Tettamanti, T.: Variable speed limit control for the motorway-urban merging bottlenecks using multi-agent reinforcement learning. Sustainability **15**, 11464 (2023)
64. Buşoniu, L., Babuška, R., De Schutter, B.: Multi-agent reinforcement learning: an overview. In: Kacprzyk, J., Srinivasan, D., Jain, L.C. (eds.), Innovations in Multi-agent Systems and Applications—1, vol. 310, pp. 183–221. Springer, Berlin, Heidelberg (2010)
65. Sharma, P.K., Zaroukian, E.G., Fernandez, R., Basak, A., Asher, D.E.: Survey of recent multi-agent reinforcement learning algorithms utilizing centralized training. In: Pham, T., Solomon, L., Hohil, M.E. (eds.), Artificial Intelligence and Machine Learning for Multi-domain Operations Applications III, (Online Only, United States), p. 84, SPIE (2021)

Knowledge Management, Decision-Making and Information and Communication Technology: A Systematic Mapping Study

Ibtissam Assoufi, Ilhame El Farissi, and Ilham Slimani

Abstract KM combines three essential elements: individuals, procedures, and technology. The use of technology promotes the integration and development of knowledge, while procedures improve organizational structure and operations. Additionally, individuals are the primary source of knowledge and interaction within the ecosystem. This paper explores the relationship between Knowledge management (KM), decision-making (DM) and Information and Communication Technology (ICT) in the form of a systematic mapping study (SMS). KM, DM, and ICT are interconnected concepts that play a vital role in organizational performance and success. KM involves managing knowledge and information, DM involves making decisions based on that knowledge, and ICT provides the technology for managing and sharing that knowledge. These three concepts work together to improve organizational efficiency, effectiveness, innovation, and competitive advantage. The seamless fusion of knowledge management, data management, and information and communication technology can pave the way for enhanced decision-making, collaborative efforts, and well-informed choices.

Keywords Knowledge management · Knowledge management system · Information and communication technology · Multi agent system · Decision-making · Systematic mapping study · SMS

1 Introduction

Knowledge is defined as the theoretical or practical understanding of a certain subject that is obtained through education or experience. With the emergence of information technologies and the expansive reach of the internet, the significance of knowledge assets has increased as a result of their sharing and collaborative research. In response,

I. Assoufi (✉) · I. El Farissi · I. Slimani
Mohammed First University Oujda, National School of Applied Sciences, SmartICT Lab, Oujda, Morocco
e-mail: Ibtissam.assoufi@ump.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_46

institutions have been creating knowledge management systems (KMS) to facilitate the acquisition, sharing, and creation of these assets and to foster organizational learning.

2 Background

The interplay of Knowledge Management (KM), Decision-Making, and Information and Communication Technology (ICT) is a critical factor in determining the effectiveness, competitiveness, and adaptability of enterprises and organizations in modern organizational settings. The integration of knowledge management (KM), information and communication technology (ICT), and decision-making processes has become a crucial factor in determining the performance of organizations in this era of rapidly advancing technologies and exponentially growing data output.

- Knowledge Management (KM)

Knowledge Management involves the systematic and strategic handling of an organization's intellectual assets. This encompasses the creation, storage, retrieval, and dissemination of knowledge to enhance organizational learning and innovation. KM initiatives aim to harness the collective intelligence of employees, facilitating the creation of a knowledge-sharing culture that fosters continuous improvement.

- Decision-Making

Choosing a course of action from the available options is known as decision-making, and it is an essential component of organizational management. Making decisions in dynamic, complicated business environments frequently presents decision-makers with risks, uncertainty, and the requirement for quick decisions. Having access to reliable information and being able to match decisions with organizational objectives are all necessary for making effective decisions.

- Information and Communication Technology (ICT)

ICT has completely changed how businesses handle information and enable communication. ICT infrastructure is the foundation of contemporary organizational operations, ranging from sophisticated analytical tools to systems for storing and retrieving data. The effective processing of enormous volumes of information is now possible through technologies like artificial intelligence (AI), big data analytics, and collaborative platforms, which enable both knowledge management (KM) and decision-making activities.

- The Intersection of ICT, KM, and Decision-Making

Information and knowledge flow is optimized in a dynamic ecosystem created by the combination of KM, decision-making, and ICT. Good knowledge management (KM) techniques improve the quality of information available to decision-makers,

allowing them to make more strategic and informed decisions. ICT solutions facilitate knowledge management (KM) processes by offering venues for data analysis, collaboration, and knowledge sharing.

As organizations navigate an era of digital transformation, understanding the intricate relationships among KM, Decision-Making, and ICT is imperative for fostering innovation, improving operational agility, and maintaining a competitive edge. This research seeks to delve deeper into this nexus, exploring the synergies and challenges that arise in harnessing the full potential of these interconnected elements.

3 Research Methodology

An overview of a research area is provided by a systematic mapping study (SMS) to identify the type and quality of research in the field and by providing a general description of the techniques and findings of main studies. A SMS involves five steps, which are:

- Defining the research questions,
- Searching for relevant papers,
- Screening the selected papers,
- Key-wording of abstract and data extracting,
- Mapping the results.

3.1 Research Questions and Reasons

This paper's major objective is to present an overview of the research on Knowledge management, Multi agent systems and artificial intelligence that has been done between the years of 2000 to 2023. As a result, we provide four research questions together with their purposes.

1. QR1: In which year, publication channels and sources were the selected papers related to Knowledge management (KM), Multi agent systems (MAS) and artificial intelligence (AI) published? For determine the current publication trends, as well as the various sources and channels for the articles selected to publish in.
2. QR2: How does KM help in decision-making? For identify the relationship between KM and decision-making
3. QR3: What are the current trends in knowledge management or what is the most used research field in knowledge management? To discover the most popular research area in KM
4. QR4: What is the relationship between ICT (Information and Communication Technology) and knowledge management? To describe the various study types conducted in ICT and KM.

3.2 Search String

With 9652 document results:

To find out the answers to the above research questions from different relevant resources, we plan a strategy for searching scientific resources based on keywords. The Boolean AND was used to join the important parts and the Boolean OR was used to join alternative words. The finale search string was defined as followed:

("knowledge management" OR km* OR "knowledge management system" OR KMS*) AND ("multi agent system" OR MAS) AND ("artificial intelligence" OR AI OR "decision-making" OR DM OR "artificial neural network" OR "machine learning" OR "deep learning" OR "bigdata" OR "communication and information technology" OR ICT) AND (model* OR algorithm* OR technique* OR method* OR tool* OR framework*).

Science Direct, IEEEXPLORE, Web of Science, ACM, Springer and Google Scholar were the six sources of research papers for this study. These libraries provide a large number of candidate papers and index conferences, several journals, and books that deal with the subject of this study.

3.3 Study Selection

We employ several selection criteria to include the most pertinent scientific papers and to remove unrelated or marginally pertinent studies when evaluating the papers we retrieve from database searches.

The following are the Inclusion Criteria (ICs):

IC1: papers presenting an overview on the use of AI, DM and MAS techniques in KM

IC2: This specific study is a published scientific paper.

IC3: Full text is available.

In addition, we use four Exclusion Criteria (ECs):

EC1: Papers written in other languages than English and French.

EC2: Papers published before 2000.

EC3: duplicated papers.

EC4: Short papers with only (2–3 pages).

EC5: Presentations or posters.

3.4 Data Extraction: Strategy and Synthesis

To define the categories of this form, we are interested in a systematic process using keyword extraction from abstracts on a sample of primary studies. The extracted

keywords are grouped together to form a set of categories representative of the population of studies in the target area.

In some cases, when the abstracts do not provide significant keywords, the process may require reading the content of the articles to identify the categories. In addition to collecting study-specific information, meta-information such as title, authors, publication details, etc. should also be included.

After selecting the pertinent papers, then used a form to collect the pertinent information from the chosen research in order to respond to the four RQs; QR1: In which year, publication channels and sources were the selected papers related to KM, MAS and AI published?QR2: How does KM help in decision-making? QR3: What are the current trends in knowledge management or what is the most used research field in knowledge management? And QR4: What is the relationship between ICT and KM?

3.5 Threats to Validity

The primary validity risks for this study are listed below:

Bias in Study Selection: In order to cover the greatest number of primary studies from the digital libraries that were used (Science Direct, IEEEXPLORE, Web of Science, ACM, Springer, and Google Scholar), we constructed a search string that contained all the key words. Selection criteria were created to precisely match the RQs in order to avoid excluding pertinent papers.

Bias in Data Extraction: Data extraction is a critical phase in the SMS process.

4 Results

To implement SMS in this work, we have set four RQs. The results of each RQ are presented below.

4.1 Studies Selection

9652 potential papers were extracted using the search string from the 6 digital libraries. 3966 papers were rejected after applying the exclusion criteria to the titles, keywords, and finally the abstracts of the candidate papers. Then, we apply the inclusion criteria to the 5686 remaining papers to obtain the 1093 selected studies. The list of chosen articles includes all the data needed to respond to this SMS's QRs.

Fig. 1 Number of papers published per year

4.2 QR1: In Which Year, Publication Channels and Sources Were the Selected Papers Related to Knowledge Management (KM), Multi Agent Systems (MAS) and Artificial Intelligence (AI) Published?

The year in which papers related to Knowledge management (KM), Multi agent systems (MAS) and artificial intelligence (AI) were published can range from the early days of computer science and AI research up to the most recent studies and developments in the field.

The number of the chosen papers that were extracted between 2000 and November 2022 is shown in Fig. 1. 1687 of the papers were published in journals, as 299 were presented in conferences. According to Fig. 1, there were much fewer publications published before 2016 than there were between 2016 and 2022.

4.3 QR2: How Does KM Help in Decision-Making?

Knowledge management is a system for collecting, organizing, and sharing the knowledge that can help in decision making [1]. It provides the necessary information to identify potential areas of improvement, identify potential problems, and determine the best course of action. It also helps to identify potential solutions, develop strategies, and make informed decisions.

KM helps in decision making by allowing businesses to store, access, and track all of their data in one organized place. This makes it easier to access the information they need when they need it, eliminating delays and allowing for faster decision-making [2]. It helps identify critical data points and trends, as well as giving decision makers a comprehensive view of potential solutions and their associated risks. It also

allows for quick access to actionable insights that can be derived from data analysis. This efficiency can lead to higher quality decisions and improved overall results.

Social media technologies, as a serious knowledge management platform, have the potential to enhance public participation in disaster response, particularly when used within formal organizations for knowledge sharing and reuse. This leads to faster decision making and more comprehensive knowledge resources [3]. However, the effectiveness of these technologies in different crisis situations is uncertain, and lessons from traditional knowledge management systems may need to be considered as they become more widespread.

A research found that knowledge management (KM) has changed in recent years and has an impact on the decision-making process. The impact of KM on decision-making depends on the specific knowledge management system (KMS) used by a company [4]. By choosing the right KMS, a company can focus on specific aspects of the decision-making process.

4.4 QR3: What Are the Current Trends in Knowledge Management or What is the Most Used Research Field in Knowledge Management?

Here are some of the current trends in knowledge management:

- Artificial Intelligence (AI) and Machine Learning (ML): AI and ML algorithms are invaluable for information curation and filtering because they are excellent at processing and interpreting enormous volumes of data. These technologies enable knowledge workers to concentrate on the most valuable and applicable knowledge assets by automatically categorizing, tagging, and prioritizing material based on relevance. Organizations may combat information overload and make sure that pertinent knowledge is easily available by utilizing AI and ML for information curation [5]. In addition, AI and ML systems offer personalized knowledge delivery, by customizing content and recommendations to each user based on their preferences, roles, and previous interactions. Organizations can offer specialized learning materials, pertinent insights, and unique knowledge experiences by comprehending user behavior and making use of recommendation algorithms. This individualized strategy improves employee engagement, retention, and information absorption.
- Cloud-based Solutions: Cloud computing has revolutionized the way organizations manage and store information. Cloud-based knowledge management solutions are becoming increasingly popular as they provide organizations [6] with the flexibility to access information from anywhere and on any device.
- Social Knowledge Management: Social media platforms, such as blogs, wikis, and forums, are becoming an important part of organizations knowledge management strategies [7], providing employees with a collaborative environment for sharing ideas and best practices.

– Mobile Accessibility: The widespread use of mobile devices has made it essential for organizations to ensure their knowledge management systems are accessible from mobile devices, enabling employees to access information on the go.
– Knowledge Graphs: Knowledge graphs, which use a graph-based data model to represent complex relationships between entities and their properties [8], are becoming an important tool for knowledge management, providing organizations with a more intuitive way to represent and access information.
– Focus on Employee Engagement: Organizations are placing greater emphasis on involving employees in the knowledge management process [9] and ensuring they have access to the information they need to do their jobs effectively.
– Integration with Other Systems: Knowledge management is no longer a standalone system, but is being integrated with other enterprise systems, such as customer relationship management (CRM) and enterprise resource planning (ERP) [10], to provide a more comprehensive view of an organization's information.

Therefore, the main research areas in knowledge management are subject to change with the current advancements and tendencies in the field. Yet, some of the frequently examined topics include knowledge sharing, representation, discovery, transfer, creation, and innovation. It is crucial to recognize that knowledge management is an interdisciplinary field that relies on theories and concepts from various domains like psychology, sociology, management, and information systems. However, some of the commonly studied areas in knowledge management include:

- Knowledge sharing and collaboration
- Knowledge representation and modeling
- Knowledge discovery and data mining
- Knowledge transfer and organizational learning
- Knowledge creation and innovation.

4.5 QR4: What is the Relationship Between ICT (Information and Communication Technology) and Knowledge Management?

Information and Communication Technology (ICT) plays a crucial role in Knowledge Management (KM) by providing the necessary tools and platforms to store, organize, and disseminate knowledge within an organization [11]. ICT enables the creation, capture, and sharing of knowledge through digital means such as databases, document management systems, intranet portals, and social media [12]. The integration of ICT in KM helps to improve access to information, facilitate collaboration, and support decision-making. In turn, effective KM practices can enhance the performance and competitiveness of organizations [13] through the effective use of the information and knowledge available to them.

5 Discussion

This section discusses the results of the four research questions.

5.1 QR1: In Which Year, Publication Channels and Sources Were the Selected Papers Related to Knowledge Management (KM), Multi Agent Systems (MAS) and Artificial Intelligence (AI) Published?

Artificial Intelligence Science (AI) was founded in the 1950s, but developed greatly a year later 2000. From Fig. 1, it can be seen that publications greatly increased from year to year due to the fact that AI and MAS are a growing concern and are being employed by more and more in researchers, notably in the field of Knowledge management.

5.2 QR2: How Does KM Help in Decision-Making?

Knowledge management is a system for organizing and sharing information that can assist in decision-making. It provides access to necessary data for identifying areas for improvement, problems, and potential solutions [14]. By storing data in an organized manner, it enables quick access and analysis of data which allows decision-makers to make informed choices based on data rather than guesswork [15]. It also helps identify critical data points, trends, and actionable insights that can lead to improved decision-making and results.

5.3 QR3: What Are the Current Trends in Knowledge Management or What is the Most Used Research Field in Knowledge Management?

In knowledge management, the current trends include the use of artificial intelligence, machine learning, and natural language processing to automate knowledge management processes, such as document classification, information extraction, and recommendation systems. Another trend is the increasing use of cloud-based platforms for knowledge management, allowing organizations to store and share knowledge more effectively.

5.4 *QR4: What is the Relationship Between ICT (Communication and Information Technology) and Knowledge Management?*

Information and Communication Technology (ICT) and Knowledge Management (KM) are closely related as ICT is the main enabler of KM. KM involves the systematic gathering, organizing, storing, and sharing of an organization's knowledge and information [16]. ICT provides the tools and technology to support this process, including databases, intranet systems, collaboration software, and social media platforms. By leveraging ICT, organizations can improve the efficiency and effectiveness of their KM processes [17] and ensure that the knowledge and information is accessible to those who need it. In essence, ICT and KM complement each other in creating an environment that supports the creation, sharing, and utilization of knowledge.

Recent research in the field has shown that enhancing Information and Communication Technology (ICT) skills greatly contributes to the growth of Knowledge Management (KM) within an organization, resulting in a long-lasting advantage over competitors [18, 19].

6 Conclusion

The goal of this Systematic Mapping Study (SMS) was to provide a summary of how decision-making (DM) and Information and Communication Technology (ICT) are utilized in the context of Knowledge management. This paper discussed the results of the four RQs. The results related to the research question (RQ) are as follows: for RQ1, after 2000, publications have increased significantly over time as AI and MAS are becoming more widely used by researchers, particularly in the area of Knowledge management. For RQ2, Knowledge management is a decision-making support system. It helps decision-makers to make informed choices based on data rather than assumptions, identifies critical data points, trends, and actionable insights, which lead to improved decision-making and outcomes. For RQ3, current trends in knowledge management involve the use of knowledge-sharing platforms to capture and share knowledge, AI to automate the process of organizing data, collaboration tools to promote teamwork, personalized knowledge management systems to cater to individual user needs, and knowledge analytics tools to gain insights into knowledge management processes for optimization. For QR4, ICT and KM are closely related, with ICT being the primary facilitator of KM. KM involves the systematic collection, organization, storage, and sharing of an organization's knowledge and information. ICT provides the necessary tools and technology to facilitate this process, such as databases, intranet systems, collaboration software, and social media platforms, it is generally agreed that ICT enables and provides the entire infrastructure and tools to support KM processes within an enterprise. To succeed in KM, it is important to assess and define ICT capabilities properly as it supports and facilitates KM.

References

1. Abubakar, A.M., Elrehail, H., Alatailat, M.A., Elçi, A.: Knowledge management, decision-making style and organizational performance. J. Innov. Knowl. **4**(2), 104–114 (2019). https://doi.org/10.1016/j.jik.2017.07.003
2. Känsäkoski, H.: Information and knowledge processes as a knowledge management framework in health care: towards shared decision making? J. Doc. **73**(4), 748–766 (2017). https://doi.org/10.1108/JD-11-2016-0138
3. Yates, D., Paquette, S.: Emergency knowledge management and social media technologies: a case study of the 2010 Haitian earthquake. Int. J. Inf. Manage. **31**(1), 6–13 (2011). https://doi.org/10.1016/j.ijinfomgt.2010.10.001
4. Nicolas, R.: Knowledge management impacts on decision making process. J. Knowl. Manag. **8**(1), 20–31. https://doi.org/10.1108/13673270410523880
5. Al-Emran, M., Mezhuyev, V., Kamaludin, A.: An innovative approach of applying knowledge management in m-learning application development: a pilot study. Int. J. Inf. Commun. Technol. Educ. **15**(4), 94–112 (2019). https://doi.org/10.4018/IJICTE.2019100107
6. Noor, A.S.M., Younas, M., Arshad, M.: A review on cloud based knowledge management in higher education institutions. Int. J. Electr. Comput. Eng. **9**(6), 5420–5427 (2019). https://doi.org/10.11591/ijece.v9i6.pp5420-5427
7. Kašcelan, L., Bach, M.P., Rondovic, B., Durickovic, T.: The interaction between social media, knowledge management and service quality: a decision tree analysis. PLoS One **15**(8), 1–30 (2020). https://doi.org/10.1371/journal.pone.0236735
8. Ji, S. et al.: A Survey on Knowledge Graphs: Representation, Acquisition, and Applications, pp. 1–21 (2021)
9. Mubarak, K., Samantha, T.: The Role of Employee Engagement on Knowledge Management and Worker Productivity: a Case Study in Sri Lanka", Sabraz Nawaz SAMSUDEEN. J. Asian Financ. **8**(4), 507–0515 (2021). https://doi.org/10.13106/jafeb.2021.vol8.no4.0507
10. Liew, C.B.A.: Strategic integration of knowledge management and customer relationship management. J. Knowl. Manag. **12**(4), 131–146 (2008). https://doi.org/10.1108/13673270810884309
11. Gutterman, A.S.: Information and Communication Technologies for Knowledge Management (2020)
12. Briones Peñalver, A.J., Santos, J.A.C., Bernal Conesa, J.A., Custódio Santos, M.: Influence of cooperation and collaborative ICT in knowledge management. J. Sci. Ind. Res. (India) **77**(6), 313–317 (2018)
13. Ramadan, B.M., Dahiyat, S.E., Bontis, N., Al-dalahmeh, M.A.: Intellectual capital, knowledge management and social capital within the ICT sector in Jordan. J. Intellect. Cap. **18**(2), 437–462 (2017). https://doi.org/10.1108/JIC-06-2016-0067
14. Intezari, A., Gressel, S.: Information and reformation in KM systems: big data and strategic decision-making. J. Knowl. Manag. **21**(1), 71–91 (2017). https://doi.org/10.1108/JKM-07-2015-0293
15. Skyrme, D., Amidon, D.: Managing extracted knowledge from big social media data for business decision making. J. Knowl. Manag. **1**(1), 27–37 (1997)
16. Sitarski, K.: The role of information technology systems in knowledge management. Found. Manag. **2**(1), 117–132 (2010). https://doi.org/10.2478/v10238-012-0024-9
17. Jacobson, M.: The importance of ICT on knowledge management in organizations. Bloomfire (2021), [Online]. Available: https://bloomfire.com/blog/knowledge-management-in-banking/
18. Ghabban, F., Selamat, A., Ibrahim, R.: New model for encouraging academic staff in Saudi universities to use IT for knowledge sharing to improve scholarly publication performance. Technol. Soc. **55**, 92–99 (2018). https://doi.org/10.1016/j.techsoc.2018.07.001
19. Foote, A., Halawi, L.A.: Knowledge management models within information technology projects. J. Comput. Inf. Syst. **58**(1), 89–97 (2018). https://doi.org/10.1080/08874417.2016.1198941

What Measurement Scales for Assessing e-reputation? A Systematic Literature Review

Mariem Hakim, Catherine Ghosn, and Razane Chroqui

Abstract The e-reputation of companies is attracting significant attention among researchers in the field of management sciences. Its analysis and measurement rely on the Internet that allows any user wishing to express their opinion to become a stakeholder involved in shaping a company's reputation. However, researchers often overlook the online dimension when developing measurement scales for e-reputation. This study aims to address this specific issue by conducting a systematic literature review to identify what is known and unknown in the current literature. The findings highlight the originality of two studies that incorporate the online dimension into their scales, enabling the measurement of e-reputation. These studies provide researchers with the opportunity to adapt existing measurement scales to the context of digital social networks and allow managers to assess their e-reputation. This paper also underscores the lack of studies specifically focused on e-reputation measurement. Therefore, there is a need for further research in this domain to fill this gap and gain a comprehensive understanding of methods for evaluating e-reputation.

Keywords Measurement scale · e-reputation · Digital social networks · Validity · Reliability · Systematic literature review

M. Hakim (✉)
University Paul-Valéry Montpellier 3, Montpellier, France
e-mail: mariem.hakim@etu.univ-montp3.fr

C. Ghosn
University Paul Sabatier, Toulouse 3, France
e-mail: catherine.ghosn@iut-tlse3.fr

R. Chroqui
Hassan First University of Settat, National School of Applied Sciences, Interdisciplinary Laboratory of Applied Sciences, Settat, Morocco
e-mail: chroqui.razane@uhp.ac.ma

A. Idrissi (ed.), *Modern Artificial Intelligence and Data Science 2024*, Studies in Computational Intelligence 1166, https://doi.org/10.1007/978-3-031-65038-3_47

1 Introduction

Since the appearance of e-reputation through the Internet, numerous studies have attempted to define and measure it [1]. However, the scales used in the current literature do not consider the online dimension of the Internet and its users as one of the company's stakeholders. This study aims to explore this scientific gap by proposing a systematic review. It is a methodology that involves in-depth research, careful selection, rigorous and reproducible analysis. The systematic review also proposes a methodical presentation of relevant scientific writings to highlight key concepts related to a topic or a theory [2, 3]. To accurately present the existing scales and identify the gaps in the literature concerning the topic addressed in this study, we will follow the guidelines of Prisma (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) and incorporate the recommendations provided by [4]. These researchers propose an approach in five steps for conducting a literature review in the field of management science: (1) Definition, (2) Research, (3) Selection, (4) Analysis, and (5) Presentation.

2 Defining the Research Question

A clear and precise research question is essential for a systematic review. An excessive specificity or excessive generality can cause problems when interpreting the results. To formulate a clear question, we will use the PICO model (Population, Intervention, Comparison, Outcome) proposed by [5] and cited by [6]. This model helps define the key elements of the question addressed to guide the systematic review.

Population (P): companies appearing on digital social networks.

Intervention (I): the use of a measurement scale to assess the e-reputation of companies on digital social networks.

Comparison (C): the existing measurement scales of e-reputation for companies.

Outcome (O): a selection of the most appropriate measurement scale for the e-reputation of companies on digital social networks.

Based on this model, the proposed research question is as follows: what measurement scale could be used for assessing the e-reputation of a company on digital social networks?

3 Inclusion and Exclusion Criteria

Inclusion and exclusion criteria are the key elements in conducting a systematic review. They serve to define the characteristics of the studies that will be included and excluded from the analysis. Inclusion and exclusion criteria are established in advance and applied systematically and rigorously.

The inclusion criteria are:

- Studies that have developed or validated measurement scales for e-reputation.
- Studies that have used these scales to evaluate corporate e-reputation.
- Studies published in peer-reviewed academic journals.
- Studies written in French or English.
- Clear methodology and information on the psychometric properties of the scale.

The exclusion criteria are:

- Non-academic articles such as blogs, magazines, or newspapers.
- Studies published in a language other than English or French.
- Studies that do not evaluate the validity and reliability of the proposed measurement scales.
- Studies without clear methodology, research question and abstract.
- Studies that address reputation in an offline context.

4 Article Search Strategy

The search for articles was conducted from November 2022 to February 2023. The selection of published studies was performed in four electronic scientific databases: Scopus, Web of Science, Cairn, Hal Archives SHS. We believe that these databases provide sufficient interdisciplinary coverage of our main theme. Grey literature was also included in the search to avoid publication bias and achieve maximum comprehensiveness. Finally, additional publications were manually searched, such as works recommended by peers. In our search strategy, six keywords were chosen (e-reputation, online reputation, measurement scale, reliability, validity, digital social networks). This choice is based on current literature and exchanges with professionals and colleagues with expertise in the fields of management and information and communication sciences, particularly regarding e-reputation issues. Limits related to the year of publication were not imposed. Moreover, to avoid noise or silence in formulating our query, we mobilised Boolean operators "and/or" by retaining three combinations:

- "e-reputation" AND ("measurement scale" OR "reliability" OR "validity") AND ("digital social networks");
- "Measurement scale" AND "reliability" AND "validity" AND "digital social networks";
- ("e-reputation" OR "online reputation") AND ("measurement" OR "measurement scale") AND ("validity" OR "reliability") AND ("digital social networks").

5 Article Selection

Article selection aims to identify relevant publications that can address the research question. Therefore, the strategy involves selecting relevant publications based on the concepts included in the research question and the inclusion criteria. In this step, we ensured that each article makes a significant contribution to this systematic review. To respect this meticulous approach, only peer-reviewed publications were considered to ensure the validity of the results. An initial screening of articles was carried out based on their titles and abstracts to determine whether they meet the inclusion or exclusion criteria. Subsequently, a second screening was performed based on the full-text articles.

6 Data Extraction and Analysis

In Table 1, the application of the PRISMA guidelines is presented. The guidelines provide a structured approach for conducting and reporting systematic reviews. The table outlines the various steps followed in the review, data analysis and extraction.

7 Results

In response to our formulated query and following the interrogation of the data selected, our systematic literature review yielded eleven studies. Of these, nine were excluded based on the predefined criteria, and we present them subsequently in the article. The retained articles, chosen for their alignment with the research objectives, form the core of our analysis.

While presenting the results, our research contributes to the current academic knowledge not only by displaying the selected articles but also by pinpointing a noticeable gap in the existing scales. We assert that this gap provides a further exploration for researchers and managers to capture and measure with precision their e-reputation.

(A) Summary of excluded articles

In Table 2, the articles excluded after full-text reading are presented. The AMAC index [7] used to be the only ranking used to measure the reputation of American companies until 1997. This model was criticized for its lack of precise definition and solid theoretical basis for these eight categories [8]. Challenging the AMAC index for its validity, [9] emphasized that the eight dimensions did not capture all facets of a company's reputation. To address these weaknesses, [9] proposed The Reputation Quotient, which considers stakeholders' sentiments toward the company while retaining the dimension of financial performance. While this model represents

Table 1 Prisma guidelines

Identification	References identified by database search (n = 54)	Additional references identified by other sources (n = 2)
Selection	Selection References after removal of duplicates (n = 47)	
	References selected based on title (n = 20)	References excluded based on title (n = 27)
Eligibility	Full-text articles evaluated for eligibility (n = 11)	Full-text articles excluded (n = 9)
Inclusion	Studies included in qualitative synthesis (n = 2)	

a significant advancement by considering multiple stakeholders, it has been criticized for placing excessive importance on the first dimension, "emotional appeal." This dimension is supposed to be a consequence of reputation rather than a facet of it. Reference [10] introduced the "Customer-Based Reputation" scale, aiming to provide improvements over previous scales, notably by incorporating customer opinions. However, some researchers like [11] argue that this scale is not applicable due to certain items. For example, for a customer to judge how employees are treated, they would need to study or have close knowledge of the organization. Reference [12] argues that the CBR scale by Walsh and Beatty [10] has certain limitations. Following Churchill's paradigm [13] they propose a nineteen-item scale across five distinct dimensions specifically designed to evaluate the reputation of large service organizations. A new scale called RepTrak Pulse is proposed by [14]. This is a simpler, more concise scale, based primarily on stakeholders' emotions, representing both a strength and a weakness of this scale. In 2004, [6] attempted to measure reputation through employees' and consumers' perceptions of the company in a dimension called "corporate character." [15] also proposed a scale based on literature, focus groups, and interviews, using weighted indicators to reflect their importance in the measurement scale. We also encounter the scale proposed by [16], consisting of four dimensions measured by twenty-one items to evaluate reputation while considering different stakeholders. In the same vein, there is also a four-item scale developed by [17] that relies on the overall impression of internet users regarding the company.

Finally, it is important to highlight that these scales were not included as they were all developed in an offline context and have not been tested online. These scales focus mainly on employees and consumers as stakeholders of the company with no item addressing the online dimension.

(B) Summary of the selected studies

To answer the question of this research, we selected two studies presented in Table 3 that provide an original contribution to the measurement of online reputation. First, the online reputation scale [1] stands out as the first scale focused on the customer in the field of marketing. Second, [18] scale is pioneering by creating the first measurement scale for e-reputation. After a quantitative study, sixteen items and four dimensions were retained: "brand characteristics, quality of website, quality of service, social media". By including the last dimension, the scale captures the influence of social media platforms on e-reputation with six items: "activity of the community, influencers' opinion on the web, buzz, attendance on social networks, number of fans/followers/tweets and number of views" (Table 3).

8 Discussion

The reference [18] proposes the very first scale for measuring e-reputation in a global online context. The authors argue that it is crucial to assess e-reputation while also acknowledging the importance of the conventional aspects of reputation. This scale

Table 2 Articles excluded after full text reading

Study n°	Scale	Title of the article	Dimensions	Items	Context	Publication	Stakeholders
1	Amac Index (Hutton, 1986)	America's Most Admired Companies		8	Offline	Fortune	Multiple stakeholders
2	Reputation Quotient (Fombrun et al., 2000)	The reputation Quotient SM: A multi-stakeholder measure of corporate reputation	6	20	Offline	Journal of Brand Management	Multiple stakeholders
3	The scale of Helm (2005)	Designing a Formative Measure for Corporate Reputation		9	Offline	Corporate reputation review	Consumers
4	RepTrack Pulse (Ponzi et al.,2011)	Reptrack Pulse: conceptualizing and validating a short-form measure of corporate reputation		4	Offline	Corporate reputation review	Multiple stakeholders
5	The scale of Davies and al. (2004)	A Corporate Character Scale to Assess Employee and Customer Views of Organization Reputation	7	12	Offline	Corporate reputation review	Employees and customers
6	The scale of Schwaiger (2004)	Components and Parameters of Corporate reputation—an Empirical Study	4	21	Offline	Schmalenbach Business Review	Multiple stakeholders
7	Customer Based Corporate reputation of large service organizations (Wepener and Boshoff, 2014)	An instrument to measure the customer-based reputation of large service organizations	5	19	Offline	Journal of Services Marketing	Multiple stakeholders

(continued)

Table 2 (continued)

Study n°	Scale	Title of the article	Dimensions	Items	Context	Publication	Stakeholders
8	The Scale of Highhouse and al. (2009)	Examining corporate reputation judgments with generalizability theory		4	Offline	Journal of Applied Psychology	Multiple Stakeholders
9	Customer Based reputation CBR (Walsh and Beatty, 2009)	The customer-based corporate reputation scale: replication and short form	5	15	Offline	Journal of Business Research	Customers

Table 3 Studies included in the final synthesis

Study n°	Scale	Title	Dimensions	Items	Context	Publication	Stakeholders
1	Customer-based online reputation scale [1]	Online reputation Scale development	6	3	Online	Academy of Marketing Science	Customers
2	Dutot and Castellano [18]	Designing a measurement scale for e-réputation	4	16	Online	Corporate reputation review	Multiple stakeholders

is validated and reliable, following Churchill's recommendations, and it consists of four dimensions: "brand characteristics, website quality, service quality, and social media". These dimensions are measured using fifteen items, among others there are the number of subscribers, fans, views, and the activity of the community manager, etc. The methodology used for validating this measurement scale involved a sample of 186 participants. The authors evaluated the dimensions of the scale to ensure their validity and reliability. Construct validity was assessed using Alpha's Cronbach, with all values exceeding 0.7, including the overall α of 0.855 for the fifteen items. The average variance explained (AVE) obtained after removing one item is 0.632 which is satisfactory. The loading scores for the remaining items were all above 0.6, with only one item falling below 0.7. Multicollinearity was assessed using the variance inflation factor (VIF), which showed values below 1.8 for all dimensions, indicating no significant collinearity. Finally, a Principal Component Analysis confirmed the contribution of all 15 items to their respective dimensions.

Reference [1] scale, on the other hand, also follows Churchill's (1979) solid methodology. It is specifically developed for customers in an online context but does not consider all stakeholders. This scale includes six dimensions: customer orientation, good employer, reliable and financially strong company, product and service quality, price, and social and environmental responsibility. The dimension of "product and service quality" consists of three sub-facets, namely reliable delivery, innovativeness and singularity, and high-standard offerings. The results of this scale validation indicate satisfactory reliability (greater than 0,7) and convergent validity (equal to or greater than 0.5). Moreover, the findings demonstrate a significant negative relationship between e-reputation and perceived risk, as indicated by a path coefficient of -0.462 ($p < 0.000$) and a bootstrapped R^2 of 21.4%. These results establish the nomological and operational validity of the scale.

In addition to the studies that have focused on developing measurement scales with items for assessing e-reputation, [19] proposed specific formulas. Their research primarily relies on user engagement and mood as determining factors for e-reputation on Facebook using three metrics: user's popularity (likes), commitment (comments), and virality (shares). For popularity, the measures evaluate the number of posts liked, average likes per post, and popularity among stakeholders. Commitment is assessed with the number of posts commented on, average comments per post, and

commitment among fans. Virality, the third aspect, is measured by metrics capturing the number of posts shared, average shares per post, and virality among fans.

The limitations of these studies lie in the lack of testing by external researchers or managers on the scales developed by the original researchers. This raises questions about the robustness, reliability, and generalizability of the scales and the results obtained. Further research endeavors should consider conducting additional testing and validation of the scales to ensure their effectiveness across various settings, platforms, and populations.

9 Conclusion

This article proposes a systematic review of the literature with a rigorous methodology in selecting scales measuring e-reputation in an online context, notably digital social networks. We emphasize that the two studies were selected based on their methodological qualities and the flexibility of items appropriate to online stakeholders and users. The findings highlight the need for further exploration and contribution from both professionals and researchers. It is crucial to recognize the impact that information circulating within virtual communities can have on a firm's e-reputation. By delving deeper into this topic, researchers can enhance the develop effective strategies to manage and leverage reputation in the online landscape.

Acknowledgements We would like to express our sincere gratitude to Dr. Ana Bumber from the LAIRDIL laboratory, University Toulouse III Paul Sabatier, for her invaluable assistance in proofreading and correcting the English version of this article. Her expertise and attention to detail have greatly improved the clarity and accuracy of the manuscript.

References

1. Chebli, Y.: L'e-réputation du point de vue client: Proposition d'un modèle explicatif (No. hal-02537782) (2016)
2. Tranfield, D., Denyer, D., Smart, P.: Towards a methodology for developing evidence-informed management knowledge by means of systematic review. Br. J. Manag. **14**(3), 207–222 (2003)
3. Briner, R.B., Denyer, D., Rousseau, D.M.: Evidence-based management: concept cleanup time? Acad. Manag. Perspect. **23**(4), 19–32 (2009)
4. El Hilali, S., Azougagh, A.: La Revue de Littérature Systématique en Sciences de gestion (2021)
5. Richardson, A.: Fatigue in cancer patients: a review of the literature. Eur. J. Cancer Care **4**(1), 20–32 (1995)
6. Davies, G., Chun, R., da Silva, R.V., Roper, S.: A corporate character scale to assess employee and customer views of organization reputation. Corp. Reput. Rev. **7**, 125–146 (2004)
7. Hutton, C.: America's most admired companies. Fortune **119**(1), 16–22 (1986)
8. Sobol, M.G., Farrelly, G.E., Taper, J.S.: Shaping the corporate image: an analytical guide for executive decision makers (1992)

9. Fombrun, C.J., Gardberg, N.A., Sever, J.M.: The Reputation Quotient SM: a multi-stakeholder measure of corporate reputation. J. Brand Manag. **7**, 241–255 (2000)
10. Walsh, G., Beatty, S.E., Shiu, E.M.: The customer-based corporate reputation scale: replication and short form. J. Bus. Res. **62**(10), 924–930 (2009)
11. Laroutis, D., Boistel, P., Badot, O.: Analyse des déterminants de la fréquence d'achat sur les sites Web marchands. Recherches en Sciences de Gestion **1**, 187–213 (2021)
12. Wepener, M., Boshoff, C.: An instrument to measure the customer-based corporate reputation of large service organizations. J. Serv. Market. (2015)
13. Churchill, G.A., Jr.: A paradigm for developing better measures of marketing constructs. J. Mark. Res. **16**(1), 64–73 (1979)
14. Ponzi, L.J., Fombrun, C.J., Gardberg, N.A.: RepTrak™ pulse: conceptualizing and validating a short-form measure of corporate reputation. Corp. Reput. Rev. **14**, 15–35 (2011)
15. Helm, S.: Designing a formative measure for corporate reputation. Corp. Reput. Rev. **8**, 95–109 (2005)
16. Schwaiger, M.: Components and parameters of corporate reputation—an empirical study. Schmalenbach Bus. Rev. **56**, 46–71 (2004)
17. Highhouse, S., Brooks, M.E., Gregarus, G.: An organizational impression management perspective on the formation of corporate reputations. J. Manag. **35**(6), 1481–1493 (2009)
18. Dutot, V., Castellano, S.: Designing a measurement scale for e-reputation. Corp. Reput. Rev. **18**, 294–313 (2015)
19. Bonsón, E., Ratkai, M.: A set of metrics to assess stakeholder engagement and social legitimacy on a corporate Facebook page. Online Inf. Rev. **37**(5), 787–803 (2013)