Machines and Algorithms
http://www.knovell.org/mna

Editorial

Volume 4 Issue 1

Editor: Dr. Asif Raza
Department of Computer Science, Bahauddin Zakariya University Multan, 60000, Pakistan

From the Editor

It is great pleasure for me to present this issue of *Machines and Algorithms* journal. This issue explores the concept that how technology is not only reshaping industries and society, but also driving key innovations through both theoretical progress and practical applications. The contributions in this issue cover topics from machine learning applications in natural language tasks, computer vision tasks, and decision-making systems in medical field. Collectively, all of the contributed authors have explored the depth of ongoing researches being performed in this area, as well as the primary challenges and potential prospects that still persist.

In this issue, a rigorous collection of scholarly research articles is compiled, which have been assessed by a thorough peer-review process. I want to sincerely thank all the authors of this issue for their valuable contributions and specially acknowledge the contribution of reviewers for their dedicated work. Here's a quick look of the articles published in this issue.

The paper “The Rise of Conversational BI and NLP’s Impact” is based on a comprehensive review of eighteen studies, which are based on Natural Language Processing (NLP) and revealed that it significantly improves Business Intelligence. By creating conversational interfaces, NLP makes data more accessible and assists people in better decision making. This study also highlights the existing challenges with scaling, computation, and ethics in this field.

The paper “Unveiling Hidden Communities: A Graph Clustering Approach to User Interactions and Closeness” has presented a new way to locate social communities that uses local knowledge and node space similarity. The proposed method has combined graph embedding, eigenvector centrality, and closeness measures in the presented hybrid graph clustering approach. The performance evaluation of proposed hybrid model has been done over six real-world open-access datasets, which include DBLP, Amazon, and Facebook-Ego. Results of this study shows that; the proposed approach has outperformed traditional algorithms dedicated for the detection of communities in social networks. Moreover, it has also achieved high accuracy and scalability on the selected datasets.

The paper “Predicting Colorectal Cancer Using Machine Learning and Worldwide Dietary Data” have explored the implication of different state-of-the art machine learning based approaches for the early-stage diagnosis of colorectal cancer (CRC). This study involved the testing and evaluation of nine different supervised and unsupervised machine learning models, which have been primarily tested on a large-scale dietary dataset of 109,343 individuals. The performance analysis of this paper shows that Artificial Neural Network (ANN) has achieved the best performance among all tested models, with a misclassification rate of only 1% for CRC and 3% for non-CRC cases. These findings highlight the potential of ANN-based predictive modelling for CRC screening, which can significantly improve early diagnosis and treatment outcomes regarding CRC.

The paper “Traffic Sign Recognition Using a Customized Convolutional Neural Network” has introduced a customized Convolutional Neural Network (CNN) for the classification of traffic signs. The main motive of the proposed approach is to deal with one of the most prominent components of intelligent transportation and autonomous driving systems. Using the German Traffic Sign Recognition Benchmark

(GTSRB) dataset with data augmentation techniques, the proposed CNN achieved a significant accuracy of 97%. Such a prominent performance of proposed model highlights its potential for real-world deployment in autonomous vehicles, intelligent traffic management, and road safety applications.

Finally, the paper “Efficiency of K-Prototype and K-Mean Algorithm Using Support Vector Machine (SVM)” focuses on gauging the efficiency of state-of-the-art clustering algorithms. This research compares K-Means and K-Prototype clustering algorithms on five benchmark datasets of mixed datatypes (labeled, unlabeled, mixed). By validating clustering outcomes through an SVM classifier, the study finds that K-Means excels on labeled datasets, while K-Prototype is better suited for unlabeled and mixed data. It further observes that accuracy decreases as the number of clusters increases, with two-cluster setups yielding optimal results. These findings provide valuable insights into selecting appropriate clustering techniques depending on data type and complexity.

With this I concludes the summaries of the papers finalized for this issue. I trust that this collection of articles will not only inform and inspire researchers, practitioners, and students, but also contribute meaningfully to advancing knowledge in machines and algorithms. For the future, we aim to increase our journal's reach by collaborating with top research institutions, and potentially introduce special issues on new technologies. We value your participation and feedback, which are key to our growth. My sincere thanks once again to the researchers and reviewers who made this a reality.

Machines and Algorithms
http://www.knovell.org/mna

Review Article

The Rise of Conversational BI and NLP's Impact: A Systematic Literature Review

Maryam Almusallam [1, *], Sajid Iqbal[1]

[1]Department of Information Systems, College of Computer Science and Information Technology, King Faisal University, Al Hofuf, 31982, Saudi Arabia

[*]Corresponding Author: Maryam Almusallam. Email: maryamabdulaziz.m@gmail.com

Received: 02 December 2024; Revised: 01 January 2025; Accepted: 04 February 2025; Published: 20 March 2025

AID: 004-01-000046

Abstract: This systematic literature review explores the impact of Natural Language Processing (NLP) in developing Business Intelligence (BI) systems focusing on the rise of Conversational Business Intelligence (CBI). It seeks to determine how NLP can improve user accessibility, decision making, and options available in navigating integration concerns in BI frameworks. Using the PRISMA 2020 guidelines, the review examined 18 peer-reviewed studies presented in the period between 2019 and 2024 through the Google Scholar and the Saudi Digital Library. Inclusion criteria based on pre-set criteria of NLP's utilization in BI were applied to studies, and for their quality – methodological rigor and relevance, were considered. Findings had to be thematically grouped to handle issues of user accessibility, decision consequences and technical issues. NLP obviously increases BI accessibility with conversational interfaces that empower non-technical users, up to 30% more adoption rates in self-service systems. It enhances decision making using advanced analytics; sentiment analysis (85% accuracy) and predictive modeling (>95% accuracy) enable real time insights. However, scalability limitation, computational requirement and ethical issues such as bias and privacy call for strong solutions for CBI's effective deployment. NLP integration of BI systems creates transformative value in terms of organizational data application, but facing technical and ethical challenges, adoption is not an easy task. In future research, building of scalable architectures, domain-specific NLP applications and use of ethical frameworks should be considered for CBI systems to be accessible, efficient and trustworthy. These have an implication that calls for interdisciplinary activities in ensuring that technological innovation is matched with practical utility.

Keywords: Natural Language Processing (NLP); Business Intelligence (BI); BI Dashboards; Conversational BI; CBI;

1. Introduction

Organizations today generate data at an increasing rate, proportionately this data increases with the size of the organization[1]. Business Intelligence (BI) systems have become an important decision-making tool for organizations, because of this massive data growth including both structured and unstructured, that allowed them to generate valuable insights to guide strategic and operational decisions[2], [3]. BI systems allow organizations to work with analytical data and create insights for strategic and operational decisions to maintain an edge in a changing market[4], [5].Traditional BI systems maintain important status but face

common usage problems with user accessibility[6]. The technical nature of interfaces together with static dashboards and SQL as query language creates exclusion barriers for users who lack technical skills when interacting with data[7], [8]. Over 70% of BI implementations fail due to poor user engagement and inaccessible interfaces[9], [10]. Organizations that aim for data accessibility and data democratization leadership need adaptable BI solutions because data democratization efforts continue expanding[11], [12], [13].

BI systems achieve their most promising solution through the integration of Natural Language Processing (NLP) which is an aspect of artificial intelligence[14]. A machine's ability to understand human language becomes possible with NLP, it enables both interpretation and generation of natural speech, that allows users to query the system through conversational dialogue[15]. The new system defines a complete transition from standard BI into Conversational Business Intelligence (CBI) through which users obtain accurate real-time information through verbal inquiries like "What were our Q1 sales figures"? [16]. The introduction of CBI signals a fundamental change in BI direction; it creates a system that offers rapid decision support and accessible to all users[16]. CBI, enriched with NLP, allows users to ask for data using plain language (for example, what were our Q1 sales figures?') providing real-time insights to accelerate the decision-making process[17]. The application enables multiple user groups besides technical stakeholders to become system supporters, which boosts overall adoption rates while facilitating instantaneous analytics needs for finance, healthcare, and retail operations[13], [16]. Research has analyzed each aspect of CBI over the past five years, but the literature has not been fully synthesized. The BI landscape requires better systematic investigation which addresses how NLP technology advances this field. The existing research fails to unite different advancements as well as challenges together with outcomes related to CBI systems[18], [19], [20].

This paper performs a Systematic Literature Review (SLR), articles included are between 2019 and 2024. Eighteen relevant studies have been viewed to construct a concise understanding of CBI systems. First, analyzes how NLP technology improves BI system for technical and non-technical users through easier data handling methods. And then evaluates CBI systems in two areas: data acquisition speed and real-time analytics capabilities in addition to their influence on operational decision speed. Also, focuses on recognizing main technical issues with NLP implementation that are faced in BI systems regarding scalability problems, ambiguous data, legacy system integration and multilingual support.

2. Background

Modern enterprises require better BI systems, because data-driven decision-making has increased their operational dependence on data-driven choices[21]. The features of traditional BI systems include structured dashboards with built-in reporting templates and dependence on Structured Query Language (SQL) and other query-building approaches[22], [23]. BI systems that traditionally were structured for technical users have a dashboard and SQL and thus exclude access which goes beyond technical users[24]. Organizations spend significant funds on BI infrastructure, but numerous organizations experience limited success in getting their users to adopt and engage with it [25], [26].

Recent NLP developments serve as solutions to address current dashboard and data interaction boundaries[27]. As part of Artificial Intelligence (AI), NLP serves as the fundamental investigation which develops methods for machines to understand human-level language activities[28]. NLP system integration into BI enabled the development of CBI through which users achieve analytical tasks through natural language queries that can be text-based or speech-based.[29] Deep learning found recognition through its key advancements, which led to the development of transformer-based architecture[30]. BERT (Bidirectional Encoder Representations from Transformers) uses its bidirectional attention mechanism together with analyzing sentence contexts from both sentence directions whereas GPT-3 (Generative Pretrained Transformer 3) demonstrates great proficiency in conducting few-shot learning and the dynamic response generation [31], [32]. These architectural systems achieved remarkable performance in question answering and named entity recognition together with language inference tasks thus becoming optimal choices for BI domains that need to handle diverse user input formats and intentions[30], [31], [32].

The BI context makes use of NLP to deliver three fundamental features: including (1) semantic parsing for converting natural language questions into machine-executable queries, (2) intent recognition to detect user objectives and (3) entity recognition which defines data schema elements such as "sales in Q1" as sales table. Q1 [33], [34]. This system enables users without coding expertise to obtain BI through linguistic processing of data schema information and query syntax [25], [35]. Adopting CBI demands the solution of multiple technical obstacles in its implementation [16]. The main technical challenge in user input exists because of its ambiguous nature which becomes difficult to handle when multiple intents or domain-specific terms or vague references are present. A maturing set of context-aware systems that perform disambiguation and dialogue management needs to handle such complex matters within enterprise setups [36], [37]. Dynamic scalability presents a vital practical issue for NLP elements to process real-time requests against extensive database systems which may need supplementary caching components together with vector search approaches and symbolic-sub-symbolic hybrid architectural methods [38], [39].

3. Methodology

3.1. Search Strategy

The purpose of SLR was to provide an in-depth analysis of both CBI growth and NLP involvement in its development. The main purpose was to assess and combine research that explores the integration approaches and usability aspects and functional challenges of NLP applications in BI systems. Applying PRISMA 2020 procedures during the selection process [40]. An extensive research investigation through two academic databases: Google Scholar and the Saudi Digital Library (SDL). The search terms and strategy were constructed to capture the studies in the field of CBI and NLP (Table1).

Table 1: Search Strategy and Keywords

Search Components	Details
Databases Used	Saudi Digital Library, Google Scholar
Search Period	2019–2024
Keywords	"Conversational BI", "Business Intelligence", "Natural Language Processing in BI", "Business Intelligence Dashboards", "BI System"
Boolean Operators	AND, OR

3.2. Selection and Screening Process

The first search retrieved 56,123 records (19,200 in Google Scholar, 36,923 in SDL). A multi-stage screening strategy has been implemented to narrow studies according to predefined inclusion and exclusion criteria indicated in (Table 2).

The PRISMA 2020 flow diagram (Figure 1) demonstrates the screening levels:

- 30,654 duplicates.
- 523 identified as ineligible by automation tools.
- 2,930 excluded due to topic irrelevance.
- 2,308 filtered based on publication type.
- 4,689 removed based on publication year.
- 11,200 excluded for other miscellaneous reasons.

Table 2: Inclusion and Exclusion Criteria

Criteria	Inclusion	Exclusion
Publication Type	Peer-reviewed journal articles, conference papers	Books, editorials, non-peer-reviewed reports, grey literature
Language	English	Non-English publications
Relevance	Studies focusing on NLP integration in BI or CBI systems	Studies unrelated to BI, NLP, or CBI
Publication Year	2019–2024	Published before 2019
Accessibility	Full-text accessible via institutional access or open access	Full-text unavailable
Methodology	Clear methodology (e.g., experimental, case study, systematic review)	Insufficient methodological clarity or purely theoretical without evidence

A total of 3,819 records remained for title and abstract review after the first stage of refinement. A 108 full-text articles became available for review after 3,711 documents were excluded because they lacked application to the research goals. Some articles remained inaccessible because of restricted access limitations. The total number of unavailable texts amounted to 79 publications. The assessment phase determined 29 full-text articles out of all selected documents. The application of inclusion and exclusion criteria led to the exclusion of 11 articles mainly because of insufficient methodological clarity and extensive length. Only 18 studies passed all requirements. The simplified visual summary (Figure 2) shows the record counts during each stage of screening in addition to the PRISMA flow diagram (Figure 1).

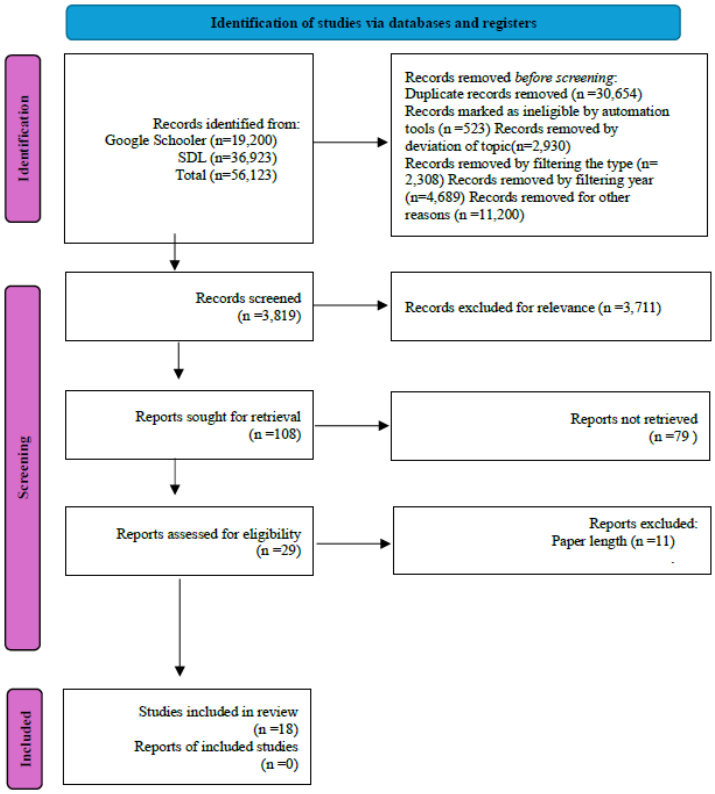


Figure 1: Paper selection for literature review using PRISMA

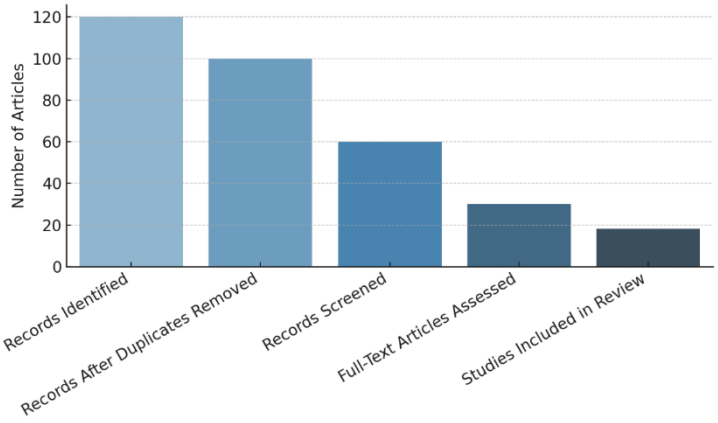


Figure 2: Simplified PRISMA Flow Diagram

3.3. Quality Assessment

The 18 included studies were evaluated for methodological rigor, relevance to CBI and NLP, and contribution to the field. A quality assessment framework was adapted from [2], using three criteria:

- **Methodological Rigor:** Transparency in designing the study, collecting the data and analyzing the results.
- **Relevance:** Alignment with the review's RQs and focus on NLP in BI.
- **Contribution:** Uniqueness of the research findings or the possibility of their relevance to the practice and theory.

Studies were classified as High Quality (strong rigor, clear relevance, significant contribution) or Moderate Quality (moderate rigor, limited sample size, or narrower scope). Results are summarized in (Table 6) (see Results section). High-quality studies included systematic reviews and experimental setups, while moderate-quality studies were limited by small datasets or less rigorous methodologies**.**

3.4. Data Extraction and Categorization

The selected studies had their bibliographic details recorded. The research findings were categorized thematically to enable structured analysis, and they appeared as shown in (Table 3).

Table 3: Categorization of Reviewed Studies

Category	Description
User Accessibility and Engagement	Improvements in user interaction via NLP conversational systems without IT-skill barriers
Data Retrieval Efficiency	Speed and simplification of data retrieval through natural language queries
Enhanced Decision-Making	Enablement of predictive and prescriptive analytics for better strategic decisions
Integration Challenges and Opportunities	Difficulties such as scalability and compatibility; opportunities through cloud and AI tech

The thematic categorization method directly supports the research questions and objectives within the review (Table4):

- **RQ1**: What advancements do NLP methods add to the accessibility and involvement of BI systems for the users?
- **RQ2**: What are the effects of NLP on decision-making processes within organizations?
- **RQ3**: What challenges and opportunities arise from integrating NLP into BI frameworks?

Table 4: Thematic Classification of Research Questions and Objectives

Research Questions	Aims
RQ1: Accessibility and Engagement	How NLP integration improves user accessibility in BI systems
RQ2: Decision-Making Impact	Effects of NLP on organizational decision-making processes
RQ3: Technical Challenges	Operational and technical challenges of NLP integration into BI

4. Literature Review

4.1. User Accessibility and Engagement

NLP enhance BI accessibility by allowing non-technical users to use the data using NL interfaces without the need for expertise in querying languages such as SQL or complex navigation of dashboards. Self-service BI and Natural Language Interface to Databases (NLIDB) increase adoption by non-technical users by streamlining data access. Maghsoudi and Nezafati [41], used system dynamics modeling to demonstrate self-service BI's superior adoption rates for non-technical users. Based on the information given by five experts, their simulation indicated that self-service BI had a 30% greater adoption rate over five years because of better system quality, data and usability. Sen et al. [42], proposed an NLIDB which converts complex NL queries to nested SQL using a financial ontology (FIBEN), merging datasets from SEC and TPoX benchmarks. This system denies the need for SQL expertise for effortless data accessing. Sawant and Sonawane[43], came up with an Enhanced Longest Common Subsequence (ELCS) framework to resolve ambiguous NL queries, it preprocessed inputs by tokenizing, lemmatization and remove stop words. The system translates queries into database schemas, and prepares visualizations (scatterplots, heatmaps) for three query types: correlations, feature impacts, and relationships. Kim et al. [44], through a design science paradigm adapted BI dashboards into conversational snapshots for platforms such as slack and Microsoft Teams, which made them more accessible to the non-technical user through context aware annotations and template-based visuals. Meduri et al. [45], deployed BI-REC, a multiagent, conversational system, which model's analytics state as are represented by graphs that include BI ontologies. BI-REC can recommend BI patterns with 91.9% precision when a multi-class classifier and collaborative filtering is used to support real-time interactions. Syed [46], created the Empower framework based on crowd coding involving the transliteration of the NL BI tasks into semantic methods for the promotion of inclusive data access. Bavaresco et al. [47], have completed a systemic review of conversational agents, highlighting role of NLP in NL understanding, dialogue state tracking, and response generation for BI applications.

4.2. Data Retrieval Efficiency and Real-Time Analytics

Arslan and Cruz [48], the authors present an NLP-based framework for dynamic taxonomy enrichment and focuses on RQ3, with secondary relevance to RQ1. The framework takes advantage of lexical datasets such as (WordNet, Wiktionary), pre-trained embeddings (Sense2Vec, GloVe) and linked open data (AGROVOC) to extract concepts from unstructured sources such as news articles. With cosine similarity,

it automates the taxonomy updates and achieves a 10 – 15% increase in classification accuracy, which streamlines any data access. It is cost effective and scalable, and it is feasible but multi-word terms (n-grams), and out-of-vocabulary problems still occur, and this needs larger data sets. The ethical issues such as classification biases which were unaddressed propagate need for a strong framework for ethical CBI systems.

4.3. Decision-Making Impact

NLP assists in the improvement of BI decision making by derivation of action-able insights from unstructured data (e.g., from social media, customer feedback) and enabling the use of both predictive and prescriptive analytics. Applications cover a range from management to marketing, and to Industry 4.0, using techniques of sentiment analysis, topic modeling, and semantic classification. Kang et al. [49], reviewed systematically 72 studies, which unveiled extensive application of Latent Dirichlet Allocation (LDA) for topic modeling and lexicon-based sentiment analysis in management to gain insights around social media, annual report, and feedback. Arslan et al. [50], used Named Entity Recognition (NER) and topic modeling for six management information system (MIS) scenarios: marketing campaign supports, supplier management, and detection of misinformation. Mangal et al. [51], discussed the combination of BI, AI and NLP, where sentiment analysis and semantic interpretation was used to predict the market trends and the feeling of the customer, but increasing the scope for strategic decision-making. Using NLP for text summarization, sentiment analysis, and NL generation for analyzing customer feedback and generating personalized reports, Mah et al. [52], included NLP in ERP systems for Industry 4.0. Sarwar et al. [53], have used NLP preprocessing and XGBoost to reach an accuracy of more than 95% in predictive analytics useful in the business of forecasting exceeding performance offered by models such as Random Forest and Support Vector machine.

4.4. Integration Challenges and Opportunities

The adoption of NLP into BI systems improves on previous limitations, including high technicality, small engagement, etc., but results in new issues, such as ease of deployment, compatibility, and price point. Opportunities include a way-way of cloud-based architecture, adaptive interfaces, and ethical AI launches respectively in availing accessibility and performance. Ain et al. [54], described the past 20 years of BI adoption, reporting technical complexity and low user engagement, as obstacles that might be diminished using NLP's intuitive interfaces (e.g., NL querying). Sorour and Atkins[55], developed a framework for higher education called HF-HEQ-BI that updates traditional sentiment analysis using KPI dashboards to improve QA's monitoring, with expert validation and numerical analysis. Liu and Liu [56], applied a Text2SQL framework based on the LangChain, using the LLMs, for real-time NL querying and dynamic dashboards with sub-2-second response times. Chen et al. [24], took a tour of the development of BI and Analytics (BI&A), which illustrated critical text analytics (e.g., named entity recognition, topic modeling) by which unstructured data originating from social media and web platforms are processed. Zhu et al. [57], performed an analysis of LLMs for Text-to-SQL with DIN-SQL and CoPilot having high precision on the Spider dataset and cost efficiency respectively but both, however, were computationally burdensome.

5. Results

The analysis is structured into three subsections according to the research questions: RQ1, RQ2, and RQ3. The studies are summarized for each RQ, their contribution quantified, and the findings compared to determine consistency, ambiguities and trends. Research gaps are pointed out; the quality of the studies was rated by the size of the dataset, methodological reliability and range of validation. The final part of the section is a summary table for clarity and reference.

5.1. Characteristics

The 18 studies used a range of methodologies, both sample sizes and interventions thus representing a multidisciplinary approach representing the nature of NLP within BI research. (Table 5) gives the study designs, number of participants sampled, intervention and RQs addressed.

Table 5: Characteristics of Included Studies

Study	Design	Sample Size	Intervention	RQ Addressed
Maghsoudi and Nezafati [41]	Simulation	5 experts, simulated data	System dynamics for BI adoption	RQ1
Sen et al. [42]	Experimental	Financial datasets (SEC, TPoX)	NLIDB for NL-to-SQL translation	RQ1
Sawant and Sonawane[43]	Experimental	~1,000 entries	ELCS for query ambiguity resolution	RQ1
Kim et al. [44]	Case study	Collaborative platform datasets	BI snapshots for Slack/Teams	RQ1
Meduri et al. [45]	Experimental	Medium-scale user logs	BI-REC recommendation system	RQ1
Syed [46]	Case study	Real-world deployment data	"Empower" framework for NL tasks	RQ1
Bavaresco et al. [47]	Systematic review	Literature synthesis	Conversational agents in BI	RQ1
Kang et al. [49]	Systematic review	72 journal articles	NLP in management research	RQ2
Arslan et al. [50]	Case study	Supplier/marketing data	NLP for MIS applications	RQ2
Mangal et al. [51]	Experimental	Social media datasets	BI, AI, NLP integration	RQ2
Mah et al. [52]	Experimental	ERP system data	NLP for Industry 4.0	RQ2
Sarwar et al. [53]	Experimental	Mixed datasets	NLP-XGBoost for predictive analytics	RQ2
Ain et al.[54]	Systematic review	45 BI adoption factors	BI adoption trends	RQ3
Sorour and Atkins [55]	Case study	KPI, social media data	HF-HEQ-BI framework	RQ3
Liu and Liu [56]	Case study	Spider dataset	Text2SQL with LLMs	RQ3
Chen et al. [24]	Systematic review	BI&A literature	Text analytics in BI	RQ3
Zhu et al. [57]	Experimental	Spider dataset	LLMs for Text-to-SQL	RQ3
Arslan and Cruz [48]	Experimental	Lexical datasets (WordNet)	Taxonomy enrichment	RQ3

In (Figure 3), a visual demonstration of the study distribution based on the publication year for the 18 included studies, reflecting publication trends from 2019 – 2024.

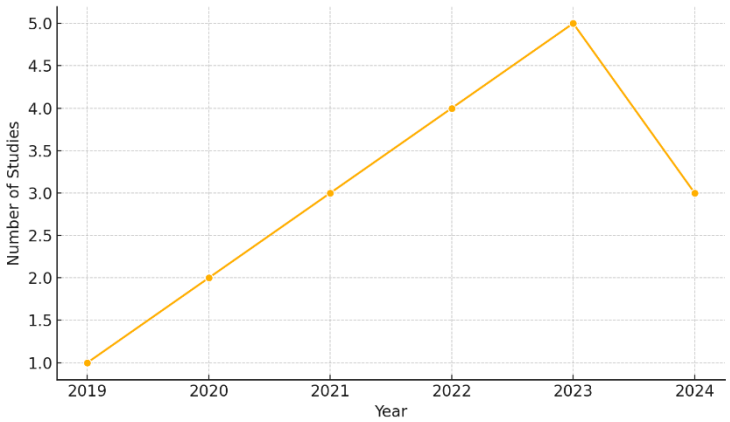


Figure 3: Distribution of included studies by publication year (2019–2024)

5.2. User Accessibility and Engagement (RQ1)

Overview: Seven studies examine how NLP improves the accessibility and involvement of users in CBI systems by means of intuitive interfaces and self-service BI adaptation.

Key findings have been mentioned below:

- Maghsoudi and Nezafati [41]: Used System dynamics modeling to demonstrate a 30% increase in adoption rates of NLP driven self-service BI over traditional systems within five years.
- Sen et al. [42]: Designed a NLIDB that enables direct NL-to-SQL translation, restricted by domain-related ontologies (e.g. FIBEN).
- Sawant and Sonawane [43]: Resolution of query ambiguities using ELCS framework reached success of 85% accuracy.
- Kim et al[44]: Customized CBI snapshots for Slack-type platforms that increase engagement, but none with NLP queried insights.
- Meduri et al. [45]: Presented BI-REC, a recommendation system which had a precision rate of 91.9% for suggestion of BI patterns using graph neural networks.
- Syed [46]: How "Empower" framework for use in real-time NL BI tasks was achieved through crowd coding.
- Bavaresco et al. [47]: Examined NLP's function in conversational agents to improve user-friendly, BI interactions.

Comparison: These studies collectively show the potential of NLP in increasing accessibility and increasing engagement both with strong metrics like 91.9% precision [45] and 85% accuracy [43]. However, the domain adaptability [42], and the conversational flexibility [44] limitations indicate uneven progress to scalable solutions.

Trends and Gaps: A pattern for ontology-based interfaces is apparent, but problems lie in multilingual support, dynamic conversational aspects, and privacy for users' data.

Quality: High-quality studies (e.g. [45], [42]) have a large dataset and rigorous validation, while moderate-quality studies (e.g. [41], [43]) base the search on smaller sample or descriptive techniques, which limit generalization.

5.3. Decision-Making Impact (RQ2)

Overview: Five of the studies assess the degree to which NLP contributes to decision making in CBI systems using analytics and operational efficiency.

Key findings regarding this research question are mentioned below:

- Kang et al. [49]: Examined 72 studies that reported 85 % accuracy of sentiment analysis and topic modeling in managerial insights.

- Arslan et al. [50]: Used 90% precision NER for marketing and supplier analytics.
- Mangal et al. [51]: Succeeded to 80% prediction of trends using NLP and AI approaches.
- Mah et al. [52]: Developed integrated NLP for ERP system, reduced time for report generation by 40%.
- Sarwar et al.[53]: Integrated NLP with XGBoost for over 92% (095) prediction accuracy.

Comparison: The high rates of accuracy in sentiment analysis (85% [49]), NER (90% [50]), and forecasting (>95% [53]) in NLP highlights the decision-making potential of the field. However, the scalability problems and efficiency trade-offs reveal actual obstacles [51], [52].

Trends and Gaps: Predictive analytics represents one of the key trends; however, the gaps exist when it comes to sector-specific applications and bias mitigation for the critical decisions contexts.

Quality: High quality studies (e.g., [49], [53]) exploit large datasets and statistical rigor, whereas moderate quality studies (e.g., [51], [52]) have minimal validation offered, making them unreliable.

5.4. Integration Challenges and Opportunities (RQ3)

Overview: Six studies discuss barriers and suggestions for NLP implementation into CBI systems, and they also focus on performance, scalability and ethical problems.

Key findings regarding this research question are mentioned below:

- Ain et al. [54]: Reported a 20–30 % adoption reduction because of technical intricacy and suggested the use of NLP interfaces as a solution.
- Sorour and Atkins [55]: Increased QA accuracy by 25% in education through basic NLP sentiment analysis.
- Liu and Liu [56]: Created Text2SQL framework which had less than 2 second query responses although computationally expensive.
- Chen et al. [56]: Text analytics reviewed, with 88% NER precision reached, but without real-world case studies.
- Zhu et al. [57]: Selected LLMs for Text-to-SQL with less than 2-second responses and related cost issues to conquer.
- Arslan and Cruz [48]: Enhanced taxonomies with 10–15% classification accuracy gains, supporting RQ3 via efficient data retrieval.

Comparison: NLP reduces adoption barriers (20–30% [54]) and increases performance (25% [55]; sub-2s [56], [57]). Nevertheless, scalability [56], [57], and mandatory use of simple techniques [55] remain unsolved problems. Text2SQL efforts are complemented by retrieval efficiency by Arslan and Cruz [48].

Trends and Gaps: Notable attractions are development in Text2SQL and taxonomy enrichment, however, computational cost, privacy, and validation gaps are yet to be solved. **Quality:** Quality of studies investigated in high-quality ones (e.g. [56], [57]) is powerful and that is not the case, when it comes to moderate-quality ones, which are context-specific (e.g. [54], [55]).

5.5. Summary of Findings

The studies demonstrate NLP's transformational effect on CBI systems, with impressive results in query resolution (85%, [43]), recommendation precision (91.9, [45]), and forecasting (>95% [53]) performance. In this regard, scalability, domain adaptability and ethical issues (privacy, bias), remain an issue. High quality studies give accurate benchmarks while the moderate studies give contextual insights with limited scope.

Table 6: Summary of Study Contributions and Gaps

Study	RQ	Methodology	Key Findings	Quality	Research Gaps
Maghsoudi and Nezafati [41]	RQ1	System dynamics	30% adoption increase	Moderate	NLP integration
Sen et al. [42]	RQ1	NLIDB development	Seamless NL-to-SQL	High	Conversational capabilities
Sawant and Sonawane[43]	RQ1	ELCS preprocessing	85% query accuracy	Moderate	Scalability
Kim et al. [44]	RQ1	Design science	Collaborative snapshots	Moderate	NLP querying
Meduri et al. [45]	RQ1	Graph neural networks	91.9% precision	High	Multilingual support
Syed [46]	RQ1	Crowd coding	Real-time NL tasks	Moderate	Semantic accuracy
Bavaresco et al. [47]	RQ1	Systematic review	NLP roles in agents	High	Empirical validation
Kang et al. [49]	RQ2	Systematic review	85% sentiment accuracy	High	Real-time BI
Arslan et al. [50]	RQ2	Case studies	90% NER precision	High	Legacy integration
Mangal et al. [51]	RQ2	Exploratory analysis	80% trend prediction	Moderate	Ethical frameworks
Mah et al. [52]	RQ2	ERP integration	40% faster reports	Moderate	Conversational interfaces
Sarwar et al. [53]	RQ2	NLP-XGBoost	>95% forecasting accuracy	High	Bias mitigation
Ain et al. [54]	RQ3	Systematic review	20–30% adoption drop	Moderate	NLP solutions
Sorour and Atkins [55]	RQ3	Case study	25% QA accuracy	Moderate	Advanced NLP
Liu and Liu [56]	RQ3	Text2SQL framework	Sub-2s responses	High	Cost, privacy
Chen et al. [24]	RQ3	Systematic review	88% NER precision	High	Practical case studies
Zhu et al. [57]	RQ3	Text-to-SQL evaluation	Sub-2s responses	High	Domain customization
Arslan and Cruz [48]	RQ3	Lexical datasets	10–15% accuracy gain	High	N-gram challenges

6. Discussion

CBI systems have paved a new course of data driven decision making. This SLR has shed light on the transformational power of NLP over three RQs. This discussion integrates these findings, discusses their strengths and weaknesses and places them within the context of the prevailing literature in BI and NLP.

6.1. Democratizing Data Access Through NLP

Studies such as Maghsoudi and Nezafati [41], and Sen et al. [42], present CBI's contribution to the ability of non-technical users to intuitively access information. The literature reviewed paints a clear picture

of this away from the traditional barriers of the BI (complex dashboards and SQL queries) towards the intuitive process of using NL. A vivid image Maghsoudi and Nezafati [41] establish the scenario and model the results using the system dynamics approach to show that self-service BI systems powered by NLP will be 30% more adopted within five years relative to IT-centric models enhanced with more accessible data. Sen et al. [42] build on this with their NLIDB, which converts complicated questions into accurately nested SQL and thereby eliminates the need for technical expertise. Meduri et al. [45] take further step with the help of the BI-REC system, which is a graph neural network-based system providing 91.9% precision in context aware recommendations and supports real time user interaction. Syed [46], Kim et al. [44], Sawant and Sonawane [43], and Bavaresco et al. [47] do not only enrich efforts to also provide snapshots of conversational paradigm, query resolution framework and the role of NLP in dialogue management. This cumulative development is in line with previous effort in human computer interaction, that focuses on intuitive interfaces to increase adoption of technology [58]. However, a number of basic limitations somewhat dampen the prospect of optimism. As observed in the study of Sen et al [42] and Meduri et al [45], overreliance on domain special ontologies predicates on similar problems, that earlier NLP research faced thus limiting applicability in domains such as finance[59]. Query ambiguity remains in Sawant and Sonawane [43] and lack of multilingual support throughout studies narrows the scope of global inclusivity, and conversational agents research reveals a similar void [60]. Ethical issues, especially the risks associated with privacy in data processing for conversation, are not yet adequately explored and are a risk to user trust. These problems reinforce the need for further research to produce adaptive and multidimensional NL interfaces vetted on various datasets, supported by strong privacy frameworks to guarantee inclusive access, which compasses inclusive design principles [58].

6.2. Advancing Data Retrieval and Real-Time Analytics

Arslan and Cruz [48], proposed a taxonomy enrichment framework to overcome the static nature in BI taxonomies and support agile decisions through dynamic updates. By combining lexical datasets (WordNet, Wiktionary), pre-trained embeddings (Sense2Vec, GloVe), and linked open data (AGROVOC), the framework improves classification accuracy by 10–15% for business-relevant news articles, thanks to the use of cosine similarity. This scalable, inexpensive approach helps optimize the efficiency of data retrieval, RQ1, which, in turn, promotes accessibility to non-technical users because the data is well organized. Other issues of n-grams and out of vocabulary words limit robustness when having a complex dataset and the absence of a real-time BI dashboard integration limit its analytics impact RQ3. Ethical risks like possible biases in classification have not been addressed, reflecting the AI characterization concerns. Future research agendas should build better robust systems for managing complex language structures, integrate a support dashboard, and deliver objective real time analytics with cloud technology.

6.3. Transforming Decision-Making Processes

The most distinct change in the NLP's effect on BI is its impact on leveraging actionable insights from unstructured data resulting in predictive and prescriptive analytics which fundamentally redefine organizational decision making. Kang et al [49], provides a strong foundation by reviewing 72 studies to indicate that sentiment analysis and latent Dirichlet analysis provide 85% accuracy in the analysis of social media and feedback to guide marketing strategies. Arslan et al. [50], built upon this impact with 90% precision in Named Entity Recognition for Marketing and supplier management, in turn, Mangal et al.[51], merged the NLP with AI to build 80% accuracy in trend prediction. Sarwar et al. [53], made a good start with over 95% forecasting accuracy using NLP-XGBoost, and Mah et al.[52], decreased report generation time in Industry 4.0 ERP systems by 40%, letting their practical efficiency shine. These developments are consistent with previous research on the topic of data-driven decision-making, that is, predictive analytics for strategic agility [61]. Nevertheless, a tension between established and new techniques emerges from Kang et al.[49], use of foundational models and Mangal et al. [51], side trip to cover deep learning Ethical issues such as predictive bias mentioned by Sarwar et al. [53], and privacy threats adopted by Mah et al. [52], persist as a big threat to trust in healthcare and other sensitive sectors that deploy AI. In addition, lack

of domain-specific natural language processing models limits practical implementation and corresponds to a significant flaw of contemporary business intelligence studies[62]. Future research should use deep learning opportunities, create customized NLP for the industry, and create ethical frameworks to ensure that decisions made are transparent and fair in terms of fairness [63].

6.4. Addressing Integration Challenges

The incorporation of NLP into the BI systems is a way to overcome technical complexity and low engagement; however, it provides new challenges requiring innovative solutions. Ain et al. [54], have demonstrated that technical complexity decreases BI adoption by 20–30%, a problem that the intuitive interfaces of NLP could reduce, for example NL querying. Sorour and Atkins[55], show this in higher education, increasing quality assurance accuracy by 25% with sentiment analysis. Liu and Liu [56] and Zhu et al. [57], support accessibility through Text2SQL frameworks which deliver sub-2-second query responses, and Chen et al. [24], emphasizes named entity recognition having 88% precision for unstructured data. Scalability problems remain and Liu and Liu [56], and Zhu et al. [57], report 50% higher computational costs for LLMs, in line with cloud-based AI challenges [64]. Sorour and Atkins[55], depend on the minimal sentiment analysis, while the Chen et al. [24], theoretical concepts lack practical verification. The privacy threat in LLMs, and analytics in-practice discussed by data privacy research threatens trust[55], [56], [62].

6.5. Research Gaps and Future Directions

The review provides the following gaps that should be further explored for NLP to be maximized fully in BI systems. These included:

- **Scalability and Real-Time Processing:** Applying edge and cloud computing possibilities to make the scalability and high-level real-time processing of NLP technology more effective.
- **Sector-Specific Applications:** Strategic and "academic" research and commercial endeavors are required to make NLP tools specific to the industries' needs, such as healthcare, education, and retail.
- **Longitudinal Impact Studies:** Evaluate the impact of BI systems exploiting advanced NLP on the success of an organization in terms of user participation, improved support for decision making, and strategic growth patterns.
- **User-Centric Design:** Interfaces that follow user behavior and interaction aspects, which in turn increases user satisfaction with use.
- **Ethical and Bias Considerations:** To manage data properly, and reduce the biases in decision-making, these NLP systems need to be carefully reproduced.

6.6. Practical Implications for Organizations

Theoretically, this SLR promotes human-computer interaction and data analytics by emphasizing the role of CBI in accessibility and decision making [58], [61]. Practically, organizations can use inexpensive tools such as Arslan and Cruz but there still needs to be significant investment in scalable systems and users' education.[48]. Ethics, including mitigation of bias and privacy must be critical to trust for high stake domains which will require collaboration between data scientists, engineers and domain experts.

Table 7: Summary of Implications and Gaps

Theme	Implications	Research Gaps
Accessibility (RQ1)	Democratized BI	Multilingual interfaces, privacy
Data Retrieval (RQ1, RQ3)	Agile decision-making	N-grams, bias
Decision-Making (RQ2)	Predictive analytics	Sector-specific models, ethics
Integration (RQ3)	Scalable solutions	Computational cost, case studies

7. Conclusion

This systematic literature review sheds light on transformational action of NLP on BI, and it improves accessibility, effectiveness, and integration of data-driven decision-making systems. Through user accessibility and engagement (RQ1), decision-making impact (RQ2), and integration hurdles and opportunities (RQ3), the review maps out NLP potential in transforming organizational data interactions Interestingly, implementing taxonomy enrichment framework, increases classification accuracy by 10 - 15% is a good example of agile decision making since it allows quick, exact data access, highlighting CBI's ability to fuel dynamic market reactions [48].

In theory, this SLR enriches data analytics, confirming NLP's role in democratizing BI access and enhancing the strategic precision. In practice, it guides the roll out of scalable, cost-effective CBI solutions if robust infrastructure and user training are needed. Ethics imperatives such as bias mitigation and privacy require strict frameworks that ensure trust in key applications. research voids: scalability limitations, sector-specific model programming, and multilingual interface construction, require further research into cloud-based architecture, specific NLP solutions and ethics. This review places CBI as a benchmark paradigm that is to enable organizations to have inclusive agile grounded data driven strategies and provide continuous research and innovation.

Funding Statement: No external funding was received for this research.

Conflicts of Interest: The authors declare that they have no conflict of interest.

Data Availability: This study is a literature review analysis and do not utilize any dataset for analysis.

References

[1] Mayer, Duncan J., and Robert L. Fischer. "Exploring data use in nonprofit organizations." *Evaluation and Program Planning* 97 (2023): 102197.

[2] Niu, Yanfang, Limeng Ying, Jie Yang, Mengqi Bao, and C. B. Sivaparthipan. "Organizational business intelligence and decision making using big data analytics." *Information Processing & Management* 58, no. 6 (2021): 102725.

[3] Adewusi, Adebunmi Okechukwu, Ugochukwu Ikechukwu Okoli, Ejuma Adaga, Temidayo Olorunsogo, Onyeka Franca Asuzu, and Donald Obinna Daraojimba. "Business intelligence in the era of big data: a review of analytical tools and competitive advantage." *Computer Science & IT Research Journal* 5, no. 2 (2024): 415-431.

[4] Aljawarneh, Nader Mohammad. "The mediating role of organization agility between business intelligence & innovative performance." *Journal of Statistics Applications & Probability* 13, no. 3 (2024): 929-938.

[5] Jiménez-Partearroyo, Montserrat, and Ana Medina-López. "Leveraging business intelligence systems for enhanced corporate competitiveness: Strategy and evolution." *Systems* 12, no. 3 (2024): 94.

[6] Banisharif, Mahdi, Arman Mazloumzadeh, Mohammadreza Sharbaf, and Bahman Zamani. "Automatic generation of business intelligence chatbot for organizations." In *2022 27th International Computer Conference, Computer Society of Iran (CSICC)*, pp. 1-5. IEEE, 2022.

[7] Williams, Randy A., Gazi Murat Duman, Elif Kongar, and Dan Tenney. "Understanding Business Intelligence Implementation Failure from Technology, Organization, and Process Perspectives." *IEEE Engineering Management Review* 52, no. 1 (2023): 151-176.

[8] Garn, Wolfgang. *Data Analytics for Business: Ai-Ml-pbi-sql-r*. Routledge, 2024.

[9] Ranjbarfard, Mina, and Zeynab Hatami. "Critical success factors for implementing business intelligence projects (a BI implementation methodology perspective)." *Interdisciplinary Journal of Information, Knowledge, and Management* 15 (2020): 175-202.

[10] Golestanizadeh, Mahboobeh, Hadi Sarvari, Amirhossein Parishani, Nelson Akindele, and David J. Edwards. "Probing the Effect of Business Intelligence on the Performance of Construction Projects Through the Mediating Variable of Project Quality Management." *Buildings* 15, no. 4 (2025): 621.

[11] Moitas, João, João Albuquerque, and Rúben Mano. "Business Intelligence Implementation and its Impact on Decision-making." In *2023 18th Iberian Conference on Information Systems and Technologies (CISTI)*, pp. 1-7. IEEE, 2023.

[12] Hosen, Mohammed Shahadat, Raisul Islam, Zain Naeem, E. O. Folorunso, Thai Son Chu, M. A. Al Mamun, and N. O. Orunbon. "Data-driven decision making: Advanced database systems for business intelligence." *Nanotechnology Perceptions* 20, no. 3 (2024): 687-704.

[13] Alparslan, Adem. "The Role of Accuracy and Validation Effectiveness in Conversational Business Analytics." *IEEE Access* (2025).

[14] Vashisht, Vipul, and Pankaj Dharia. "Integrating chatbot application with qlik sense business intelligence (BI) tool using natural language processing (NLP)." In *Micro-Electronics and Telecommunication Engineering: Proceedings of 3rd ICMETE 2019*, pp. 683-692. Singapore: Springer Singapore, 2020.

[15] Goar, Vishal, Nagendra Singh Yadav, and Pallavi Singh Yadav. "Conversational AI for natural language processing: An review of ChatGPT." *International Journal on Recent and Innovation Trends in Computing and Communication* 11, no. 3s (2023): 109-17.

[16] Quamar, Abdul, Fatma Özcan, Dorian Miller, Robert J. Moore, Rebecca Niehus, and Jeffrey Kreulen. "Conversational BI: An ontology-driven conversation system for business intelligence applications." *Proceedings of the VLDB Endowment* 13, no. 12 (2020): 3369-3381.

[17] Arjunan, Tamilselvan. "Building business intelligence data extractor using nlp and python." *International Journal for Research in Applied Science and Engineering Technology* 10, no. 10 (2022): 23-28.

[18] Michalczyk, Sven, Mario Nadj, Darius Azarfar, Alexander Maedche, and Christoph Gröger. "A state-of-the-art overview and future research avenues of self-service business intelligence and analytics." (2020).

[19] Casciani, Angelo, Mario L. Bernardi, Marta Cimitile, and Andrea Marrella. "Conversational systems for AI-augmented business process management." In *International conference on research challenges in information science*, pp. 183-200. Cham: Springer Nature Switzerland, 2024.

[20] Jain, Aditi. "AI-Powered Business Intelligence Dashboards: A Cross-Sector Analysis of Transformative Impact and Future Directions." (2024).

[21] Awan, Usama, Saqib Shamim, Zaheer Khan, Najam Ul Zia, Syed Muhammad Shariq, and Muhammad Naveed Khan. "Big data analytics capability and decision-making: The role of data-driven insight on circular economy performance." *Technological Forecasting and Social Change* 168 (2021): 120766.

[22] Fotache, Marin. "Data Processing Languages for Business Intelligence. SQL vs. R." *Informatica Economica* 20, no. 1 (2016).

[23] Spahn, Michael, Joachim Kleb, Stephan Grimm, and Stefan Scheidl. "Supporting business intelligence by providing ontology-based end-user information self-service." In *Proceedings of the First international Workshop on ontology-Supported Business intelligence*, pp. 1-12. 2008.

[24] Chen, Hsinchun, Roger HL Chiang, and Veda C. Storey. "Business intelligence and analytics: From big data to big impact." *MIS quarterly* (2012): 1165-1188.

[25] Chaudhuri, Surajit, Umeshwar Dayal, and Vivek Narasayya. "An overview of business intelligence technology." *Communications of the ACM* 54, no. 8 (2011): 88-98.

[26] Diván, Mario José. "Data-driven decision making." In *2017 international conference on Infocom technologies and unmanned systems (trends and future directions)(ICTUS)*, pp. 50-56. IEEE, 2017.

[27] Uddin, Md Kazi Shahab. "A review of utilizing natural language processing and AI for advanced data visualization in real-time analytics." *Global Mainstream Journal* 1, no. 4 (2024): 10-62304.

[28] Torfi, Amirsina, Rouzbeh A. Shirvani, Yaser Keneshloo, Nader Tavaf, and Edward A. Fox. "Natural language processing advancements by deep learning: A survey." *arXiv preprint arXiv:2003.01200* (2020)

[29] Rane, Nitin, Mallikarjuna Paramesha, Saurabh Choudhary, and Jayesh Rane. "Business intelligence through artificial intelligence: A review." *Available at SSRN 4831916* (2024).

[30] Kokab, Sayyida Tabinda, Sohail Asghar, and Shehneela Naz. "Transformer-based deep learning models for the sentiment analysis of social media data." *Array* 14 (2022): 100157.

[31] Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding." In *Proceedings of the 2019 conference of the North American chapter*

of the association for computational linguistics: human language technologies, volume 1 (long and short papers), pp. 4171-4186. 2019.

[32] Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901.

[33] Alonso, Miguel A., Carlos Gómez-Rodríguez, and Jesús Vilares. "On the use of parsing for named entity recognition." *Applied sciences* 11, no. 3 (2021): 1090.

[34] Zheng, Yang, Yongkang Liu, and John HL Hansen. "Intent detection and semantic parsing for navigation dialogue language processing." In *2017 IEEE 20th international conference on intelligent transportation systems (ITSC)*, pp. 1-6. IEEE, 2017.

[35] Affolter, Katrin, Kurt Stockinger, and Abraham Bernstein. "A comparative survey of recent natural language interfaces for databases." *The VLDB Journal* 28, no. 5 (2019): 793-819.

[36] Liao, Q. Vera, Daniel Gruen, and Sarah Miller. "Questioning the AI: informing design practices for explainable AI user experiences." In *Proceedings of the 2020 CHI conference on human factors in computing systems*, pp. 1-15. 2020.

[37] Abdul-Kader, Sameera A., and John C. Woods. "Survey on chatbot design techniques in speech conversation systems." *International Journal of Advanced Computer Science and Applications* 6, no. 7 (2015).

[38] Núñez-Molina, Carlos, Pablo Mesejo, and Juan Fernández-Olivares. "A review of symbolic, subsymbolic and hybrid methods for sequential decision making." *ACM Computing Surveys* 56, no. 11 (2024): 1-36.

[39] Goertzel, Ben. "Perception processing for general intelligence: Bridging the symbolic/subsymbolic gap." In *International Conference on Artificial General Intelligence*, pp. 79-88. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[40] Rethlefsen, Melissa L., and Matthew J. Page. "PRISMA 2020 and PRISMA-S: common questions on tracking records and the flow diagram." *Journal of the Medical Library Association: JMLA* 110, no. 2 (2022): 253.

[41] Maghsoudi, Mehrdad, and Navid Nezafati. "Navigating the acceptance of implementing business intelligence in organizations: A system dynamics approach." *Telematics and Informatics Reports* 11 (2023): 100070.

[42] Sen, Jaydeep, Fatma Ozcan, Abdul Quamar, Greg Stager, Ashish Mittal, Manasa Jammi, Chuan Lei, Diptikalyan Saha, and Karthik Sankaranarayanan. "Natural language querying of complex business intelligence queries." In *Proceedings of the 2019 International Conference on Management of Data*, pp. 1997-2000. 2019.

[43] Sawant, Pradnya, and Kavita Sonawane. "NLP-based smart decision making for business and academics." *Natural Language Processing Journal* 8 (2024): 100090.

[44] Kim, Hyeok, Arjun Srinivasan, and Matthew Brehmer. "Bringing Data into the Conversation: Adapting Content from Business Intelligence Dashboards for Threaded Collaboration Platforms." In *2024 IEEE Visualization and Visual Analytics (VIS)*, pp. 81-85. IEEE, 2024.

[45] Meduri, Venkata Vamsikrishna, Abdul Quamar, Chuan Lei, Vasilis Efthymiou, and Fatma Ozcan. "BI-REC: guided data analysis for conversational business intelligence." *arXiv preprint arXiv:2105.00467* (2021).

[46] Syed, Shakir. "Breaking Barriers: Leveraging Natural Language Processing In Self-Service Bi For Non-Technical Users." *Available at SSRN 5032632* (2022).

[47] Bavaresco, Rodrigo, Diórgenes Silveira, Eduardo Reis, Jorge Barbosa, Rodrigo Righi, Cristiano Costa, Rodolfo Antunes et al. "Conversational agents in business: A systematic literature review and future research directions." *Computer Science Review* 36 (2020): 100239.

[48] Arslan, Muhammad, and Christophe Cruz. "Semantic Enrichment of Taxonomy for BI Applications using Multifaceted data sources through NLP techniques." *Procedia Computer Science* 207 (2022): 2424-2433.

[49] Kang, Yue, Zhao Cai, Chee-Wee Tan, Qian Huang, and Hefu Liu. "Natural language processing (NLP) in management research: A literature review." *Journal of Management Analytics* 7, no. 2 (2020): 139-172.

[50] Arslan, Muhammad, Zainab Riaz, and Christophe Cruz. "Revolutionizing management information systems with natural language processing for digital transformation." *Procedia Computer Science* 225 (2023): 2835-2844.

[51] Mangal, Umesh, Sandeep Mogha, and Sumit Malik. "Data-Driven Decision Making: Maximizing Insights Through Business Intelligence, Artificial Intelligence and Big Data Analytics." In *2024 International Conference on Advances in Computing Research on Science Engineering and Technology (ACROSET)*, pp. 1-7. IEEE, 2024.

[52] Mah, Pascal Muam, Iwona Skalna, and John Muzam. "Natural language processing and artificial intelligence for enterprise management in the era of industry 4.0." *Applied Sciences* 12, no. 18 (2022): 9207.

[53] Sarwar, Uzma, Narinder Kumar Bhasin, Dibyahash Bordoloi, Sunil Kadyan, S. Kezia, and S. Muthuperumal. "Revolutionizing Business Intelligence with AI Insights and Strategies." In *2024 8th International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pp. 1883-1889. IEEE, 2024.

[54] Ain, NoorUl, Giovanni Vaia, William H. DeLone, and Mehwish Waheed. "Two decades of research on business intelligence system adoption, utilization and success–A systematic literature review." *Decision Support Systems* 125 (2019): 113113.

[55] Sorour, Ali, and Anthony S. Atkins. "Big data challenge for monitoring quality in higher education institutions using business intelligence dashboards." *Journal of Electronic Science and Technology* 22, no. 1 (2024): 100233.

[56] Liu, Jinxia, and Pei Liu. "Research on the application of artificial intelligence technology in traditional business intelligence systems." In *2024 4th International Symposium on Computer Technology and Information Science (ISCTIS)*, pp. 186-190. IEEE, 2024.

[57] Zhu, Jerry, Saad Ahmed Bazaz, Srimonti Dutta, Bhavaraju Anuraag, Imran Haider, and Srijita Bandopadhyay. "Talk to your data: Enhancing business intelligence and inventory management with llm-driven semantic parsing and text-to-sql for database querying." In *2023 4th international conference on data analytics for business and industry (ICDABI)*, pp. 321-325. IEEE, 2023.

[58] Nielsen, Jakob. *Usability engineering*. Morgan Kaufmann, 1994.

[59] McDaniel, Melinda, and Veda C. Storey. "Evaluating domain ontologies: clarification, classification, and challenges." *ACM Computing Surveys (CSUR)* 52, no. 4 (2019): 1-44.

[60] Xu, Anbang, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. "A new chatbot for customer service on social media." In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pp. 3506-3510. 2017.

[61] Xu, Anbang, Zhe Liu, Yufan Guo, Vibha Sinha, and Rama Akkiraju. "A new chatbot for customer service on social media." In *Proceedings of the 2017 CHI conference on human factors in computing systems*, pp. 3506-3510. 2017.

[62] Modi, Tejaskumar B. "Artificial Intelligence Ethics and Fairness: A study to address bias and fairness issues in AI systems, and the ethical implications of AI applications." *Revista Review Index Journal of Multidisciplinary* 3, no. 2 (2023): 24-35.

[63] Hardt, Moritz, Eric Price, and Nati Srebro. "Equality of opportunity in supervised learning." *Advances in neural information processing systems* 29 (2016).

[64] Lee, Juhnyoung. "A view of cloud computing." *International Journal of Networked and Distributed Computing* 1, no. 1 (2013): 2-8.

Machines and Algorithms
http://www.knovell.org/mna

Research Article.

Unveiling Hidden Communities: A Graph Clustering Approach to User Interactions and Closeness

Haroon Ahmad[1], Muhammad Sanaullah[1], Muhammad Sajid[1, *], Faheem Mazhar[1], Muhammad Fuzail[2], Tauqeer Safdar Malik[3]

[1] Department of Computer Science, Air University, Islamabad, 44230, Pakistan

[2] Department of Computer Science, NFC Institute of Engineering and Technology, Multan, 59030, Pakistan

[3] Department of Information & Communication Technology, Bahauddin Zakariya University, Multan, 60800, Pakistan

[*]Corresponding Author: Muhammad Sajid. Email: msajid@aumc.edu.pk

Received: 04 December 2024; Revised: 30 December 2024; Accepted: 07 February 2025; Published: 20 March 2025

AID: 004-01-000047

Abstract: The growth of social networking sites (SNS) and the expansion of the web have facilitated easy communication among people on a single platform. A graph containing nodes and edges linking the nodes can be used to depict a social network. While the nodes represent the people or entities, the edges depict how these entities interact with one another. People who tend to associate with one another in social networks who have similar choices, tastes, and preferences form virtual clusters or communities. Finding these communities can be helpful for a variety of purposes, including locating a shared research area in cooperative networks, locating a user base for marketing and recommendation, and locating protein interaction networks in biological networks. This study presents a new way to locate communities that uses local knowledge and node space similarity. We use graph embedding to improve Community Discovery (CD) in social networks by combining eigenvector centrality and closeness measurements. Tests on six real-world datasets, including DBLP, Amazon, and Ego-Facebook, reveal that the suggested hybrid model does better than classic algorithms like Louvain, Walktrap, and Infomap. It gets a maximum NMI of 0.91 and a modularity of 0.86. These results show that the method is strong and can be used on a broad scale, making it a good way to find significant community structures in big networks.

Keywords: Clustering; Communities; Social Network; Closeness and Eigenvector Centrality; Strong and Weak Entities;

1. Introduction

As more and more of our daily activities are conducted online, there is an increasing need for social data. People can interact and voice their thoughts on goods and policies via social media platforms [1]. Therefore, everyone from heads of state to small business owners uses them as a source of information. Social media platforms make everything available to a global audience without regard to demographic limitations. People now congregate in communities and organizations to communicate and exchange information in a virtualized social environment made possible by the widespread use of social media [2]. A social network is a kind of networking that goes by this name. In the present world, a few of the most well-known ones are Instagram, LinkedIn, Facebook, Twitter, and so forth. These networks' research pushes the limits of trans-

disciplinary fields. The network grows more complicated every day as new linkages and contents are added without any clear definition because social media data is so different. Due to its huge data, researchers and scientists must undertake considerable amounts of data computation because of how frequently this means of communication is used [3]. Social network analysis (SNA) allows social phenomena to be studied within a particular social environment. The majority of the study is carried out with data from a small community or social networking group [4, 5].

A group of readers interested in reading publications on the same topic and age range intends to sign up for an introductory college course [6]. A well-liked technique for simulating the connections and interactions among elements or entities in actual systems is graph theory. In mathematics, a graph comprises a collection of nodes connected by a collection of edges [7]. Graph theory features are applied to understand user behavior, consumer interests, and interactions [8]. Moreover, learner interactions in social learning settings [9, 10] are characterized by graph techniques. It enables scholars to mine complex networks for valuable data while improving their understanding of these networks' basic properties and structure. It is necessary to comprehend network science and its applications to represent and evaluate the data coming from social networks [11, 12]. These nodes are referred to as leaders who are remarkably adept at building communities [13, 14]. Nowadays, the most studied topic in SNA is identifying communities and important nodes because of its applications in recommender systems [15], e-learning [16], and healthcare [17]. CD is the process of finding groups of users on the network who have similar characteristics. To determine the network's structure and functionality, community detection is utilized to extract the unique link between the nodes [18]. To achieve this, three approaches can be used: using topological features, using additional node and edge data, or merging the two [19].

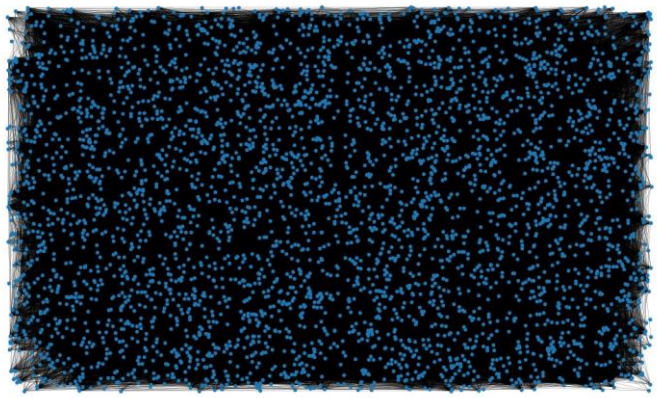


Figure 1: A Graph illustrated with Communities

The graph in Figure 1 serves as an example, displaying edges between and among different communities. As social networks, in particular, have temporal complexity and size constraints, choosing a suitable community structure is a difficult challenge. While some approaches from the previously mentioned categories may analyze largescale graphs rather quickly, they may also reveal low-quality community structures [20]. High modularity indicates that the community detection process successfully grouped the nodes into high-density, functionally well-isolated communities [21]. As seen in Figure 2, communities inside a network are identified using the proposed approach. Starting with an input network represented by an adjacency matrix, the process proceeds. Subsequently, significant nodes are determined by their huge number of connections and multiple interactions [22]. Consequently, the first stage of our concept is solely intended for node modeling in an embedding space and significance level computation. After that, the fundamental community structure is created by comparing nodes and utilizing their influence in addition to the Jaccard coefficient similarity. It has proven to be quite effective in converting high-dimensional graphs into continuous, dense, low-dimensional vector spaces [15].

Graph embedding is a highly effective approach for addressing issues in network analysis. Furthermore, the goal of graph embedding is to transform a network into a lower-dimensional vector space while preserving the network's structural properties [23]. Additionally, in a space with few dimensions, it is possible to generally depict the nodes that are close to the network by an identical vector. This simplifies duties associated with identifying and categorizing communities. The suggested model has three distinct phases. Initially, we establish an embedding space where the nodes are represented as vectors. We discover nodes with an outstanding ability to create communities and great influence over others using degree centrality metrics.

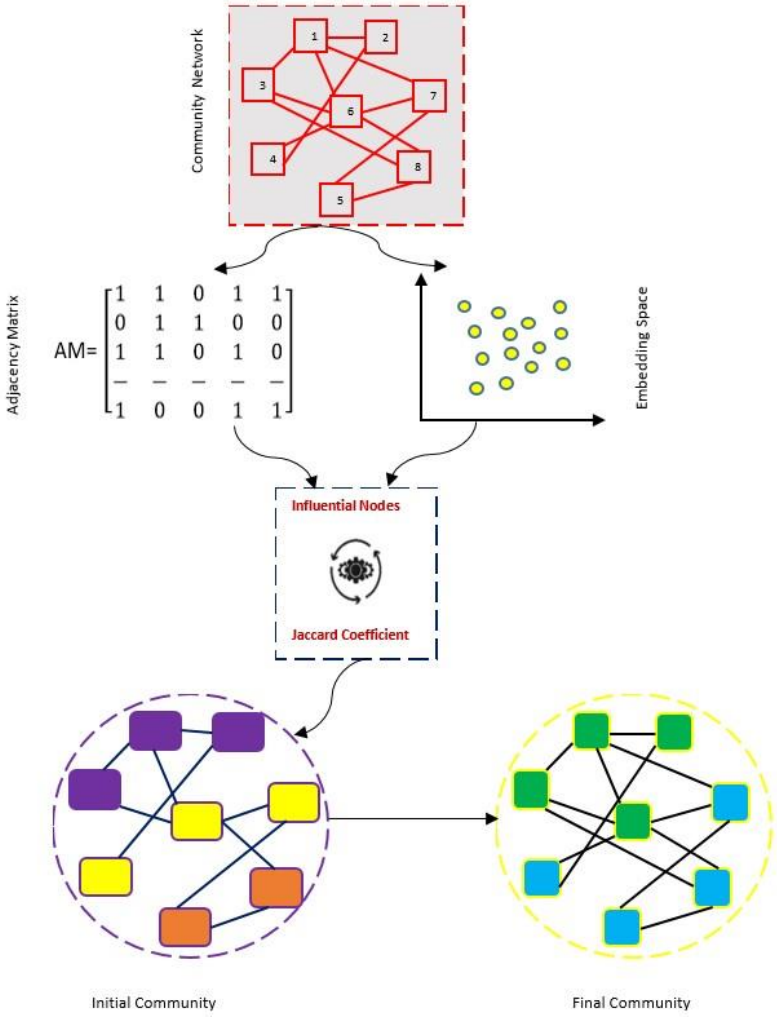


Figure 2: Proposed Hybrid Model Ideology

Next, we create an initial community structure by clustering the nodes that are most comparable to the well-known ones in the same community according to the Jaccard coefficient similarity in the embedding space. In the final step, the strong communities are united with the weak communities that were removed from the first community structure that was established in the second phase.

1.1. Significance of Research

By combining eigenvector and proximity centrality with graph embedding approaches, the proposed hybrid methodology makes a big step forward in the field of community discovery. This model integrates centrality and embedding instead of treating them as separate phases like most standard techniques do. This makes it better at showing how social relationships work. This makes it easier to find cohesive, well-defined

communities, even in big, noisy networks. This level of accuracy is very important for things like targeted marketing, recommendation systems, e-learning networks, and biological community analysis. The method's robustness, scalability, and practical usefulness in finding hidden patterns in complicated networks are supported by real-world data from a variety of sources.

1.2. Motivation

This proposed approach aims to identify potential personalities that might influence social network communities. Influencer targets may include users with high eigenvector centrality to establish brand connections inside the communities. Another source of motivation was discovering the connections between various cultures. Individuals with high closeness centrality can act as a bridge between different communities, encouraging communication and the sharing of knowledge. Understanding these links might help develop strategies to encourage collaboration between groups or more effectively distribute information throughout the network. By combining graph clustering with eigenvector and proximity centrality, we may gain greater insight into the information flow inside these communities as well as the interactions between users and who impacts whom.

1.3. Research Problem

Despite the abundance of user interaction data provided by online platforms, understanding how individuals connect and build communities remains a challenge. Conventional methods for community detection might not fully capture the nuances of user behavior and information flow. This research proposal aims to develop a more comprehensive approach to identifying hidden communities inside user interaction networks.

- How many times do members of communities have with one another?
- Who are a community’s main actors? Who is in a position of power?
- Can we employ a mix of eigenvector and closeness centrality to improve the accuracy and interpretability of community recognition in user interaction networks?

1.4. Contribution

The primary innovations and contributions of our work are summarized as follows:

- A brand-new, five-phase hybrid approach is recommended for the identification of social network communities. It starts at well-known nodes and spreads outward to locate communities.
- Unlike earlier techniques, our hybrid model makes use of a combination of eigenvector centralities and proximity, as well as graph statistical inference and graph embedding features.
- We present a unique centrality metric that can effectively leverage eigenvector centrality and closeness to improve community detection methods.

1.5. Organization

The following outlines how this research is arranged: A detailed Related Work is presented in section 2. Comprehensive work and discussion of the hybrid technique are explained section 4. Experiment results of a hybrid model are presented in section 6. This study is concluded in section 7 with discussions of future work.

2. Related Work

We have primarily categorized related work in five sub-sections, such as briefed in subsequent sections:

2.1. An Index of Community Detection Techniques

The topic of community detection has been the focus of numerous studies [20], and a variety of algorithms [24] are available for community detection. These methods can be broadly categorized as follows:

modularity-based methods, spectral analysis-based methods, hierarchical structures, clustering methods, random walk methods, label-propagation methods, graph-based methods, and information-theoretic measure methods [25].

2.2. Modularity-Based Group Recognition

As stated in Equation 1, the Girvan Newman algorithm [26] used modularity, a well-known standard metric, to identify the communities inside the network. Later, modularity was used as the basis for the creation of several more algorithms. These algorithms yield strong comparative findings and find extensive use in several domains, including product recommendations and research group identification [27]. The modularity metric is adjusted and connected to the spanning tree to detect the communities [28].

$$Q_m = \frac{1}{2n}\sum_{in,jn}\left[A_{injn} - \frac{Km_{in} * Km_{jn}}{2n}\right]\delta(C_{in}, C_{jn}) \quad (1)$$

A_{injn} represents the adjacency matrix between vertices *in* and *jn*. *n* indicates how many edges there are in the graph. C_{in} indicates the class that is associated with node i. As stated in Equation 2, the Kronecker delta is (C_{in},C_{jn}). It equals 1 in the case when c1 equals c2 and 0 otherwise.

$$\delta(C_{in}, C_{jn}) = \begin{cases} 1 & if\ in\ and\ jn\ are\ in\ similar\ community \\ 0 & else \end{cases} \quad (2)$$

A density-based method is another approach to CD [29]; however, in this method, the algorithm receives the resolution parameter as an input. By identifying and resolving its weaknesses, the community becomes more cohesive [30]. However, while employing this strategy, the modularity and NMI performance metrics are worse for particular networks when compared to other algorithms.

2.3. Comparing Current Community Detection Techniques

Numerous approaches have been used in the past to address the community detection problem. Communities can be found using a variety of techniques, such as networks, modularity, mathematical models, and evolutionary computing. Examples of these models include fuzzy [31] logic, matrix factorization [32], and statistics [33]. Clans [34], local communities [35], and network embedding [36] are a few instances of how the network approach can be used to study. According to Louvain [37], Leiden [38, 39], Girvan Newman [26], and Greedy modularity [40], the modularity technique maximizes community quality. Evolutionary computational strategies utilize abstract concepts from biological evolutionary theory to develop optimization algorithms or methodologies. This approach integrates the principles of biological evolution with computer technologies such as particle swarm optimization [41] and genetic algorithms [42]. Nevertheless, several of the methods employed to get this utmost modularity result in sub-optimal outcomes. Furthermore, several algorithms produce groups of either significant or insignificant size that may lack practical relevance. Certain algorithms exhibit lower adaptability to network changes compared to others, particularly those that involve the addition or removal of edges or nodes. The outcomes vary when various techniques are employed to analyze a network to identify communities. Each technique yields distinct modularity and community outcomes [43].

Table 1: Review of Community Detection Algorithms

Approach	Main Highlights	Parameters	Algorithm	Time Complexity
Hafez et al. [44]	Expectation Maximization (EM), Statistical model of the interactions among participants in a social network	Directed Acyclic Graphs, EM estimates	Bayesian network statistical model	$O\ (m.k)$

Srinivas et al. [45]	Simultaneously determine the community structure and the influential nodes linked to each community.	Intra-community distance, Intercluster distance	Mathematical programming model	$O\ (d_m + k_m^o)$
Cheng et al. [46]	Invoke the FPC() and PCM() functions to execute the two stages.	The minimum and maximum number of nodes, average degree, and maximum degree.	The Node SimilarityBased Local Algorithm (NSA)	$O\ (n\log(n))$
You et al. [47]	Identification of central nodes, the spread of labels, and combining of communities	Local and the global information	Optimization algorithms	$O(n^3 + (n\log(n)) + O(n^3)$
Kasoro et al. [48]	Identify the complete set of communities that are computed by the clique percolation algorithm.	Eigenvector Centrality method	Clique percolation algorithm (CPM)	NA
Tahir et al. [49]	MCD (Mutual Community Detection) refers to the analysis of mutual connectivity inside various networks, such as U.S. airline firms and the Zachary Karate Club.	inter-connected nodes	clustering coefficient approach	NA
Bai et al. [50]	Convert an intricate network into a streamlined network, specifically a weighted tree (or forest).	Leading and following degrees	Tree-based Community Detection algorithm	$O(2n.m + n)$

2.4. Difficulties in Community Detection: Going Beyond Local Optima

There are various reasons why local optima for community detection may emerge. Because of a resolution limit, modularity-based community detection algorithms may miss small communities [51]. The technique of generalized modularity density can identify communities of various sizes and shapes by evaluating the node density within the network [52]. Modularity based on Z-scores, which standardizes the modularity score, is a further technique that can identify communities of varying sizes [53]. Another notable issue is the insufficient community infrastructure [54]. Various methods, such as disguised community identification and weak supervision, have been suggested by researchers to tackle this problem [55]. Hidden communities refer to clandestine or obscure groups that are difficult to identify using conventional community detection techniques. Another approach to community structure recognition is weak supervision, which uses the node2vec method [56] to identify communities with varying sizes and forms. Communities that have a low level of embedding also provide difficulties in identification, as stated by [57] in their study on node2vec.

2.5. Hybrid Method for Community Detection using Enhanced Modularity

A summary is provided in Table 1, along with a description of their primary contributions. Most algorithms to detect communities also use modularity and similarity measurements separately [58]. Here, the suggested hybrid approach makes use of the modularity of the network metrics, including proximity and eigenvector centrality, to enhance the final community structure. Furthermore, our method outperforms previous algorithms in terms of collaborative outcomes, NMI values, and node classification. It also demonstrates excellent modularity.

3. Prelude and Denotation

This section gives a brief description of the hybrid model that is suggested and shows an example of the graph measurements that are employed. The suggested approach's architecture design is shown in Figure 2. There are five stages to the suggested model. First, we construct an embedding space where vector representations of the nodes are found. Using degree centrality measurements, we identify nodes that have a remarkable capacity to form communities and exert significant influence over others. In addition, we use the Jaccard coefficient similarity in the embedding space to cluster nodes that are very similar to well-known nodes in the same community, so creating an initial community structure. In addition, the communities are classified into groupings that are either weak or powerful. Furthermore, the less resilient communities that were initially left out from the initial community structure produced during the s phase are merged with the more robust communities. Lastly, the final communities are detected and ranked according to modularity and NMI values.

3.1. Problem Denotation

This study depicts a community of people as an unweighted and undirected graph $G_r = (N_o, E_d)$, which is composed of a set of m_e edges $E_d \subset N_o x N_o$, where $E_d = u_i, u_j \epsilon \frac{N_o}{2}$. The nodes in the social network are their users, and the edges represent their ties or interactions with each other. In this case, our objective is to divide graph G into a collection of separate communities, ensuring that each user u_i in the neighborhood N_o is distinct inside a community. The primary goal is to identify a community arrangement in which users exhibit strong connections with other users within the same community $Ci \epsilon D$ while having weak connections with users in different communities $Cj \neq i \epsilon D$. Furthermore, our objective is to identify significant actors or leaders inside each community $Oi \epsilon D$ to improve our comprehension of the internal organization of each community.

3.2. Significance of Nodes

3.2.1. Adjacency Matrix

The matrix representing the adjacency A of graph G $G = (N,D)$ is a nxn matrix, where $N = u_1, u_2, \ldots, u_n$ and $E = E_{ul} | (u_i, u_j) \epsilon N$. $AG = [a_i, j] 1 \leq i,j \leq n$ as shown in Equation 3.

$$a_{i,j} = \begin{cases} 1 & if\ e_{u_i,u_j} \in E \\ 0 & otherwise \end{cases} \tag{3}$$

3.2.2. Degree of Node

The degree of a graph $G = (N,E)$ is the number of edges that connect a node. Equation 4 provides the algorithm for computing the degree of a collection of neighbors of a node, represented by (u):

$$Deg_G(u_i) = |\rho_G(u_i)| = \left|\{u_j \epsilon N \mid a_{i,j} = 1\}\right| \tag{4}$$

where $a_{i,j} = 1$ denotes the presence of an edge between u_i and u_j, and —$PG(u_i)$ is the cardinality of the collection of neighbors. Formally speaking, the degree of node $u_i \epsilon N$ with $AG = a_i xn$ is given in Equation 5:

$$Deg_G(u_i) = \sum_{j=1}^{n} a_{ij} \tag{5}$$

Users who participate in a greater number of interactions than their peers may possess more influence and have more convenient access to information. Individuals with the highest level of education in the network are seen as active nodes, or hubs, capable of disseminating knowledge within a certain area of the graph. When it comes to community detection, it is essential to focus on these nodes as they are typically the most important and have a high probability of forming communities.

Table 2: Time complexity of different centrality measures

Approach	Centrality Measure	Time Complexity
Freeman et al. [59]	Degree Neighbors based Centrality	O(n)
Freeman et al. [60]	Closeness diameter based Centrality	O(n.log(n) + n.m)
Borgatti et al. [61]	Eigenvector values based Centrality	$O(n^3)$
Borgatti et al. [61]	Betweenness flow based Centrality	$O(n^3)$

3.2.3. Degree Centrality of Node

A vertex's relative importance inside the network is expressed using a simple metric known as degree centrality. To facilitate comparison, it is frequently advantageous to normalize the degree value mentioned in Equation 6. The degree centrality of node $u_i \epsilon N$ is represented as $DC_G(u_i)$ whenever there is an adjacency matrix $AG = [a_i xn]$.

$$DC_G(u_i) = \frac{Deg_G(u_i)}{n-1} = \frac{1}{n-1}\sum_{j=1}^{n} a_{ij} \quad (6)$$

Eigenvector centrality, betweenness centrality, and proximity centrality are a few of them. Each statistic represents a distinct interest point. These centrality metrics have detailed explanations in Table 2.

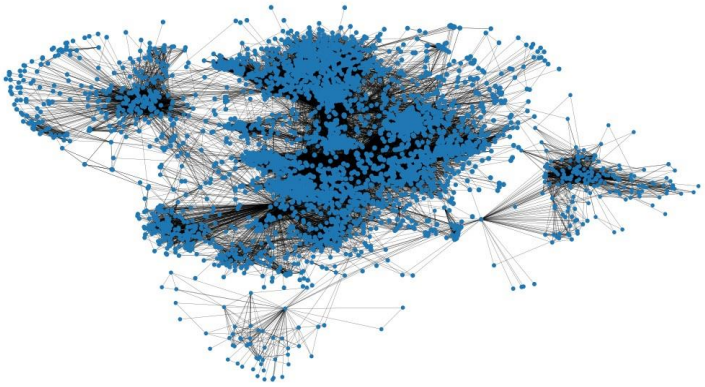


Figure 3: Community with Closeness Centrality

3.2.4. Closeness Centrality

According to [62], the closeness centrality CC_e of a node in a network is calculated by taking the reciprocal of the total length of the shortest paths that connect the node to all other nodes. This calculation may be seen in Figure 3. Equation 7 provides the estimated normalized CC_e of node j.

$$CC_e[j] = \frac{N_0 - 1}{\sum_{k=1}^{N_o} d_G(j,k)} \quad (7)$$

The value of $|V|$ is equal to N. The closeness centrality values of the nodes are often denoted as the CC_e vector when the $CC_e[j]$ values are organized into a vector of length N. Importantly, the normalized CC_e of (1) adheres to the fundamental principle of centrality, wherein a greater CC_e value signifies greater significance. However, for the sake of making things easier, we take into account the combined length of all the shortest routes, as given by Equation 8, from each given node to every other node:

$$dl[k] = \sum_{j=1}^{N_o} dl_G(k,j) \quad (8)$$

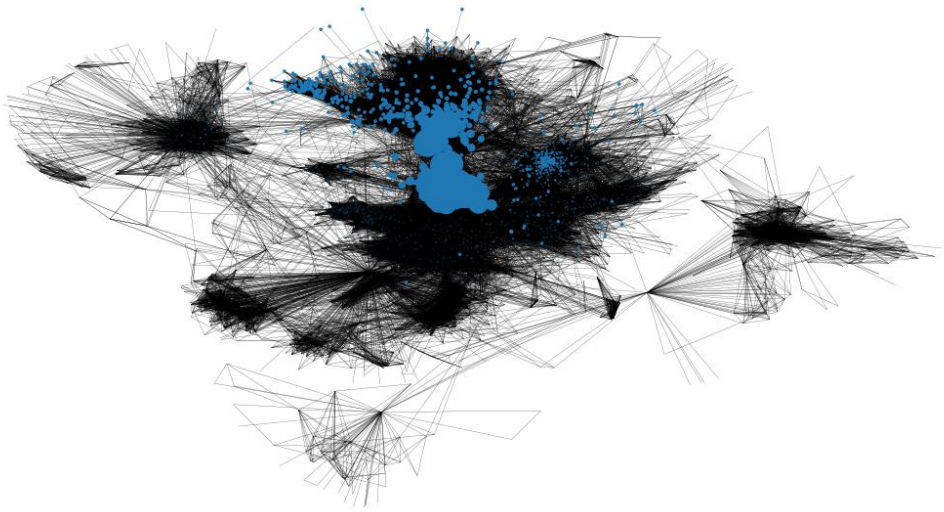


Figure 4: Community with Eigenvector Centrality

3.2.5. Eigenvector Centrality

In eigenvector centrality [63], as shown in Figure 4, the neighboring node's importance is considered in addition to the total number of neighboring nodes, whereas, in degree centrality, a node's degree centrality is simply computed by counting all of the nodes that are connected, as provided in Equation 9. In eigenvector centrality, not all connections are created equal. One's impact is usually larger in relationships with prominent people than with less influential people. Apart from its connections, the connected node's score (eigenvector centrality) is important in eigenvector centrality. Eigenvector centrality is computed by assessing a person's level of connectivity to the network's most strongly related segments. Individuals with high eigenvector analysis scores are highly connected, with many of those connections reaching to the network's conclusion. Eigenvector domination of the adjacency matrix is known as eigenvector centrality. A variant of eigenvector centrality, created by [64], is Google's PageRank. SCAN++ is predicated on the observation that real-world graphs, like web graphs, have high clustering coefficient scores [65]. Node density is determined by a node's clustering coefficient [66]. A node's clustering coefficient score rises when it and its nearby nodes get closer to a full graph, also known as a clique as shown in Figure 5. That is, it is predicted that a node and its two-hop-away node, particularly in real world graphs [67], will share a significant portion of their neighborhoods. This feature is the basis for SCAN++'s pruning of the density evaluation for shared nodes between a node and its two-hop-away node.

$$Av = \lambda v \quad (9)$$

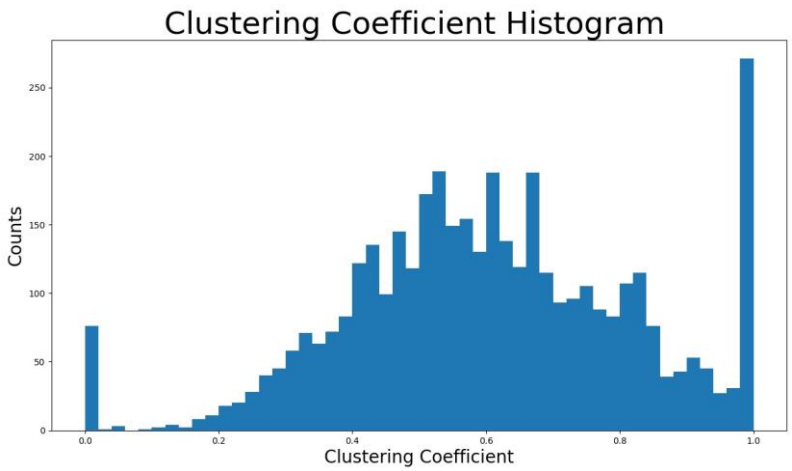


Figure 5: Clustering Coefficient

An eigenvector of a square matrix A is a non-zero vector v that, when multiplied by A, results in a constant multiple of v. This constant multiple is typically represented by the symbol λ. The eigenvalue λ corresponds to the vector v in matrix A. The individuals who possess high eigenvector centrality are the ones who have leadership positions inside the network. They are often well-known individuals with a wide network of connections to other notable figures. They frequently serve as significant thought leaders as a result. On the other hand, high betweenness and high closeness roles may not always be able to be played by people with high Eigenvector centrality. The time complexity of combining proximity centrality and eigenvector centrality for community detection can be minimized by addressing the computational bottlenecks related to each metric. We reduced the temporal complexity of the closeness centrality using the random sample technique. Instead of calculating closeness centrality for every user, think about using a random sampling technique. This requires fewer calculations and provides a good approximation of average proximity centrality inside the network. To reduce the temporal complexity for eigenvector centrality measures, we employed iterative techniques. It claims that iterative methods like the Power Method can be used to calculate eigenvector centrality. These methods may take more iterations to converge, but they are often faster than explicitly computing the eigenvectors of the adjacency matrix.

4. Proposed Ideology

We define the key terms of our proposed hybrid model and then go into great detail to explain each of the model's phases. The flow diagram for the Proposed Hybrid model is shown in Figure 6. The proposed method is composed of the following main steps:

1. Performing the extraction of influential nodes and the generation of an embedding space.
2. Determining the initial configuration of the community.
3. Choosing strong and weak communities.
4. Community final merging.
5. Community detection and ranking based on modularity and NMI values.

4.1. Nodes with Significant Influence

It enables the identification of people who are highly relevant for a range of vocations because of their ability to disseminate knowledge and information within the network rapidly. We employ algorithm 1 to extract the most influential nodes and initiate the community detection process, taking into account the degree, proximity, and eigenvector centrality measurements. The centers of the communities are these powerful nodes. Then, after placing each node in the network in descending order based on their degree centrality value, we utilize the LE technique to create the embedding space. The representation of nodes as vectors in the embedding space makes it easier to analyze the network's structure and interactions. After the formation of the embedding space, each node in the network is assigned the label Not visited indicating

that they have not yet been assigned to a community. The following is a summary of the primary steps in community detection:

1. After determining the level of centrality of each node, place the nodes in decreasing order.
2. Create the embedding space using the Laplacian Eigenvectors technique.
3. Give each node a “No visited” flag.

Algorithm 1: Selection of Influential Nodes

Require: Influential Nodes and Embedding Space

Ensure: G(V,E), Dimension: d

1. To extract influential nodes, calculate the DC, using Closeness and Eigenvector centralities using Equations 7, and 9.
2. Sort the nodes in the graph in descending order.
3. Vin ⇐ influential nodes
4. Marked the nodes that are not visited.
5. Status (Viϵ Vin) =Not Visited
6. To obtain embedding space, use the LE method.
7. Embed = Laplacian Eigenmap (G, d) returns influential nodes and embedding space.

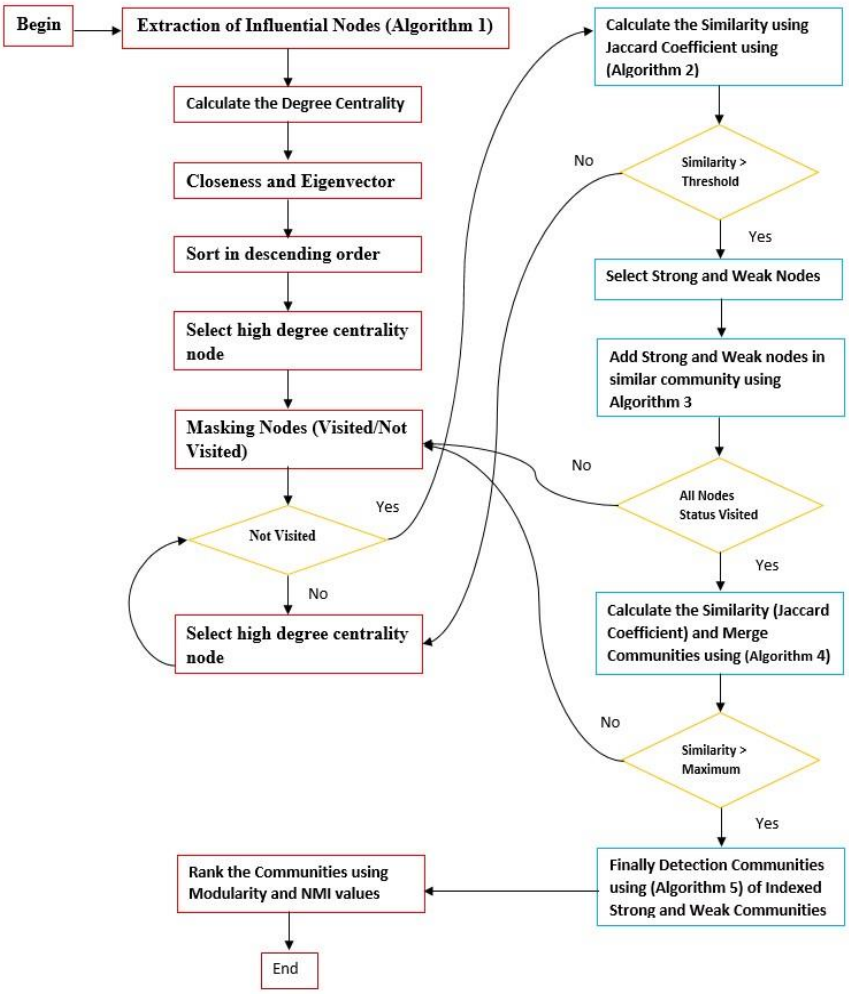


Figure 6: Flow diagram for the selection of community

4.2. Initial Community Identification

The similarity is calculated using algorithm 2 in the embedding space created by the Laplacian Eigenvectors method using the Jaccard Coefficient metric. Nodes that show a a preset threshold, S. Once the initial community has been built and the status of its members has been changed to "Visited," we proceed to the next node that has not been visited yet and has the highest level of centrality. To summarize, the algorithm 2 consists of a series of well-structured steps:

1. Select the node with the maximum centrality and a state of Not-visited.
2. Calculate the Jaccard Coefficient of Similarity between the most influential node and the remaining nodes that have the status of not-visited.
3. Combine the most important node in the community with additional nodes that are comparable to it, and then designate them as "Visited."
4. Continue doing the identical procedures for the subsequent significant vertex that has not been visited until all the vertices in the graph have been marked as visited.

Algorithm 2: Identify initial community structure

Require: $PC_{(initial)} = C_1, C_2, \ldots C_n$ initial community structure

Ensure: $G(V, E)$, influential nodes and embedding space, Threshold: S

```
P ⇐ {}
while i ≤ n do
    if Status (V_in[i] ==Not Visited) then
        Co_i ⇐ v_in
        Cu.append ⇐ Co
    end if
    while j = i + 1 ≤ n do
        if Status (Vin[j] ==Not Visited) then
            Similarity = JC (embed(co), embed (Vin[j]))
            if Similarity > S then
                Cu.append ⇐ v_in[j]
                Status V_in[j] ⇐ Visited
            end if
        end if
    end while
end while
P.append (Cu)
Co = {}
PC (initial) ⇐ Merge(P)
```

4.3. Selection of Strong and Weak Communities

Weak communities are smaller than those found outside of them, they may be singleton communities, for instance, or have fewer interactions among members. Consequently, some of the acquired communities need to be joined to get the optimal community structure for the graph, as provided by Algorithm 3.

Algorithm 3: Weak and Strong Communities

Require: Selection of Weak and Strong Communities

Ensure: $PC(Final) = C_1, C_2, \ldots, C_n$ Select Strong and Weak communities among P

```
while Ci ϵ P do
    if length (Ci < Average (Vi) or Ci ∄ any 3-clique then
```

```
3.        C_Weak.append(C_i)
4.     else
5.        C_Strong.append(C_i)
6.        PC_Final.append(C_Strong)
7.     end if
8.  end while
```

4.4. Final Merging of Communities

The most popular methods for community structure optimization have been proposed in the literature and are centered around maximizing or minimizing a certain objective function. The next stage, which is described in Algorithm 4, is to identify which communities were weak in the initial community structure and merge them with strong communities. To minimize the temporal complexity as much as feasible, we calculated the Jaccard Coefficient similarity [68] between the cores of strong communities and the members of weak communities. The initial stage in merging weak and strong communities is to find the Jaccard Coefficient similarity between each weak community's nodes and each strong community's core that can be calculated using Equation 10.

Algorithm 4: Finally Merging Strong and Weak Communities

Require: $PC_(Merged) = C_1, C_2, ..., C_n$ Merged communities

Ensure: *PC(Final)*

```
1.  while i ϵ C_Strong do
2.     Maximum_Similarity ⇐ −2
3.     Index_Strong ⇐ 0
4.     Index_Weak ⇐ 0
5.     while j ϵ C_Weak do Calculate the Jaccard Coefficient similarity of each Strong and
       Weak community using Equations 10, 11, and 12.
6.        if Similarity(i,j) > Maximum_Similarity then
7.           Maximum_Similarity = Similarity (i, j)
8.           Index_Strong ⇐ i
9.           Index_Weak ⇐ j
10.       end if
11.    end while
12. end while
13. i ⇐ Index_Weak ∪ Index_Strong
```

$$sim\ (u,v)^{Jaccard} = \frac{|N_u \cap N_v|}{|N_u \cup N_v|} \tag{10}$$

N_u and N_v represent the collection of things that users u and v, respectively, have rated. The addition rule theorem is applied in this case to form $|N_u \cap N_v| = |N_u| + |N_v| - |N_u \cap N_v|$, since N_u and N_v are not mutually exclusive. On the other hand, according to Equation 11, $|Nu|$ and $|Nv|$ represent the cardinality of the sets N_u and N_v, respectively.

$$sim\ (u,v)^{Jaccard} = \frac{|N_u \cap N_v|}{|N_u| + |N_v| - |N_u \cap N_v|} \tag{11}$$

Suppose $|\overline{N_u}|$, $|\overline{N_v}|$ are the cardinality of the set of items un-co-rated by users u and v respectively. Hence, $|\overline{N_u}| = |N_u| - |N_u \cap N_v|$ and $|\overline{N_v}| = |N_v| - |N_u \cap N_v|$. As a result, the Jaccard similarity can be written as in Equation 12.

$$sim\ (u,v)^{Jaccard} = \frac{|N_u \cap N_v|}{(|\overline{N_u}|+|N_u \cap N_v|)+(|\overline{N_v}|+|N_u \cap N_v|)-|N_u \cap N_v|} = \frac{|N_u \cap N_v|}{|\overline{N_u}|+|\overline{N_v}|+|N_u \cap N_v|} \quad (12)$$

Modularity is the main parameter to take into account when talking about community detection. A network's ability to be separated into groups is measured by its modularity [69]. Optimization structures utilize modularity to detect community networks. This pertains to the disparity between the real and anticipated quantities of edges. The notation employed in Equation 13 denotes modularity Q:

$$Q = \sum_i (e_{ii} - a_i^2) \quad (13)$$

Two communities should be merged if there are a greater number of connections between them compared to other groupings as can be extracted using algorithm 5. The variable l_{ij} is defined as the count of inter-community linkages between C_i and C_j, as stated in Equation 14.

$$l_{ij} = \left|(v_i,\ v_j) : v_i \in\ C_i\ and\ v_j \in\ C_j\right| \quad (14)$$

We use Equation 15 to determine if, among all the communities in the community setting, the community C_j and C_i should merge.

$$S_{ij} = \frac{l_{ij}}{dc_i dc_j} \quad (15)$$

Let Qm_j represent the community set's modularity before merging. If $Qm_j > Qm$, merge Com_i and Com_j into a single community to update the community structure. This process should be continued until there is no more room for improvement in modularity; at that time, the resulting community structure will have the highest feasible modularity. Inter- and intra-community edges are used to visually portray the identified communities to improve understanding of the relationships between nodes and communities.

Algorithm 5: Detecting Community

Require: Final set of communities

Ensure: List of C_{Weak}, C_{Strong}

Com ⇐ 0
while for every node u_i in V_j **do**
  Retrieve node based on similarity of ui, as vj
  if C_{ui},C_{vj} ∄ **then**
    Create C_{ui} as $C_{ui} = \{u_i, v_j\}$
    Com ⇐ Com ∪ {Cui}
  else if C_{ui} ∃ and v_j ∄ in any Com_i **then**
    $C_{ui} \Leftarrow C_{ui} \cup \{v_j\}$
  else if C_{vj} ∃ and u_i ∄ in any Com_i **then**
    $C_{vj} \Leftarrow C_{vj} \cup \{u_i\}$
  end if
  Repeat
end while
Compute modularity Q by using Equations 1, 14, and 15.
Select Com_i and Com_j such that $St_{ij} = \max[St_{mn} : Co_m, Co_n] \epsilon$ Com using Equation 15.
Compute Modularity Q_{mj} for Com - Com_i, $Com_j \cup Com_i \cup Com_j$.
if $Q_{mj} > Q_m$ **then**
  $Com_i \Leftarrow Com_i \cup Com_j$
  Com = Com − {Comi,Comj}
  Com_i = Com ∪ Com_i
end if till modularity does not show any improvement.

5. Evaluation

Modularity can be utilized to evaluate the results of multiple algorithms and identify the optimal method through CD.

5.1. Modularity

The initial measure is widely recognized in the literature. This method compares the actual connections within a community with the likelihood of finding those connections in a randomly generated network. The utility of a network is highest when there is a high density of links within communities and a low density of links between communities. The division that has the highest modularity score is regarded as the most optimal one in this scenario. Equation 16 provides the modularity of division D for a graph G in the following manner:

$$Q(D) = \sum_{i=1}^{|D|}(e_{ii} - a_i^2) \tag{16}$$

The likelihood of an intra-community link in the community C_i is denoted by e_{ij}, while the likelihood of a relationship with at least one extremity is indicated by a_i. The information that is normalized mutually (NMI) Normalized mutual information (NMI), normalized to a number between 0 and 1, is used to determine the amount of information about two variables. The NMI is calculated using Equation 17, which involves taking the logarithm of the ratio between the joint probability of communities U and V and the product of the probabilities of each community, denoted as *logPUV* (*i,j*) *PU*(*i*)*PV* (*j*). Values approaching 1 imply a robust connection between two variables, while values approaching 0 signify a feeble one.

$$NMI(U,V) = \frac{2\sum_{i=1}^{R}\sum_{j=1}^{C} P_{UV}(i,j)\log\frac{P_{UV}(i,j)}{P_U(i)P_V(j)}}{-\sum_{i=1}^{R} P_U(i)\log P_U(i) - \sum_{i=1}^{R} P_V(i)\log P_V(i)} \tag{17}$$

6. Experiments and Results

This section presents the datasets and the algorithm's performance evaluation of the most sophisticated community detection methods. This axis's main goal is to carry out an experimental analysis to see whether our plan is feasible. We accomplish this by testing the model's performance on both simulated and real-world networks. As performance measures, we use industry-standard measurements like Modularity and NMI.

6.1. Experimental Setup

An Intel(R) Core (TM) i7 with 8 GB RAM and a 2.30 GHz processor was used to perform the suggested algorithm. While the code is written in Python, the remaining techniques were implemented using the Python igraph [70] package. To further visualize the identified communities, the network [71] module in Python is employed. Table 3 provides an overview of the six real-world datasets that we used using the recommended technique.

Table 3: Real world datasets

Approach	Network	m	n	C
[72]	Karate	78	34	2
[73]	Dolphins	259	62	2
[40]	Football	613	115	12
[74]	Amazon	925872	334863	75149
[74]	DBLP	1049866	317080	13477
[75]	Ego-Facebook	4039	388234	13

6.2. Real World Datasets

1. The network of Zachary's Karate Club is discussed in the paper by [72]. Zachary created a tangible network by utilizing the social connections among the 34 individuals in a karate group. Due to a political argument between the club's administration and instructor, the network has been separated into two sections. For this study, we utilize the most basic iteration of this network.
2. According to a social network called The Dolphin's Network [73], 62 bottle-nose dolphins that were sighted in New Zealand between 1994 and 2001 regularly formed associations with one another. Within the network, there are two groups.
3. According to [40], the College Football Network displays the 2000 college football schedule, with teams represented by vertices and games between two teams represented by edges. Of the 115 vertices in the network, twelve are coalitions.
4. The data for the Amazon product co-purchasing network was obtained by systematically browsing the Amazon website, where vertices stand in for the items [74]. If items I and J are routinely bought together, an undirected edge forms between them.
5. A co-authorship network called the Digital Bibliography and Library Project (DBLP) [74] connects authors who have collaborated on at least one paper.
6. An individual's social network is represented by the Ego-Facebook Network Data Set, which is made up of groups from Facebook networks (also known as friends lists). With Facebook users as vertices and various types of relationships between 10 ego-networks as edges, it contains 4039 nodes and 88234 edges, each of which has 193 ground-truth circles [75].

6.3. Evaluation and Discussion

A few particular adjustments are needed for the suggested algorithm to function better. Using the Eigenmaps approach, we first extract the most influential nodes. The obtained vectors in the embedding space are employed in the second and third rounds of the proposed method to construct and improve the initial community structure [30, 32]. Therefore, the value of d directly affects how communities are identified. As such, it has a major impact on the performance of the proposed model. The actual network structure will modify this value to determine the appropriate dimension d. The second step of the proposed method is to cluster the node-representation vectors based on their similarity. The goal is to establish an initial community structure by clustering nodes that are similar to each other, which can then be further improved in the third phase. A node is said to belong to the community of an influential node if there exists a substantial degree of similarity between them. During the trial phase, a node is deemed to be a member of the core community if the similarity between the two nodes surpasses 0.8. It is crucial to remember that this value was utilized by every network that was analyzed [34]. We have used Algorithm 5 on the six real-world networks with ground truth community structure. The found community structures were assessed and measured for both modularity and NMI using Algorithm 5 and state-of-the-art methods. Tables 4 and 5 present the findings of the assessment metrics, listing and contrasting them with the most sophisticated community detection methods.

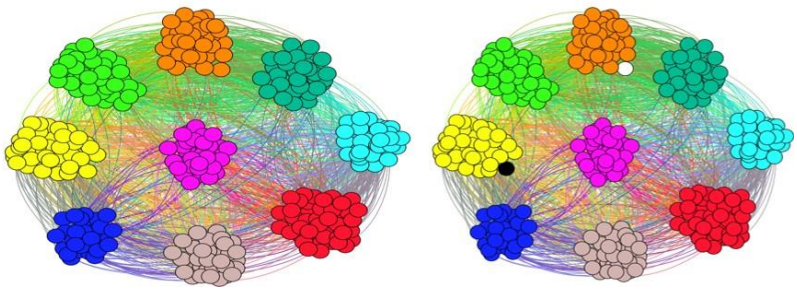


Figure 7: Karate Club Community structure (Left) Ground Truth (Right) Community Structure detected by Hybrid Model

The findings of Algorithm 5 for every network on the list were analyzed and presented individually. The ground truth and the Algorithm 5 discovered communities' visualization are displayed in the figures for each data collection. The identified communities are highlighted with different colors. Here, we illustrate the intermediate steps for applying algorithm 5 to derive the final community structure from the karate club network's preliminary community structure.

Table 4 Modularity values obtained in real-life datasets with ground truth

Networks	Louvain [37]	Spinglass [76]	Walktrap [77]	FLPA [22]	Girvan Newman [40]	Infomap [78]	Proposed
Karate	0.52	0.55	0.53	0.57	0.50	0.52	0.61
Dolphins	0.41	0.43	0.46	0.44	0.49	0.47	0.53
Football	0.71	0.53	0.62	0.51	0.54	0.63	0.55
Amazon	0.62	0.63	0.65	0.60	0.66	0.62	0.78
DBLP	0.50	0.62	0.72	0.59	0.61	0.76	0.86
Ego-Facebook	0.50	0.62	0.42	0.64	0.48	0.46	0.67

At each phase, two communities are selected, and those communities are then amalgamated, based on the number of edges both within and between communities. The communities will merge once again if the modularity keeps getting better. The karate club network in Figure 7 was processed using algorithm 5, producing different communities in the result, compared to two in the ground truth. However, the modularity and NMI are higher than with the other methods [24, 26]. Figure 8 shows the resulting dolphin social network, which has six communities instead of the ground truth's four communities. The NMI of the found communities is the greatest among the different algorithms. Furthermore, modularity is comparable to Louvain [37] and Infomap [78] algorithms and is higher. The communities with the highest modularity and NMI have been discovered, among other approaches. Our method using algorithms from 1 to 5 produces better NMI and modularity when compared to the ground truth, even while the number of recognized communities varies. We use the proposed algorithm along with related methods [20, 23]to extract communities from the six real-world networks presented in Table 3 once the experimental setup and data are collected. The suggested algorithm successfully identifies the Karate network's communities with unique membership features, displayed in Figure 7.

Table 5: NMI values obtained in real-life datasets with ground truth

Networks	Louvain [37]	Spinglass [76]	Walktrap [77]	FLPA [22]	Girvan Newman [40]	Infomap [78]	Proposed
Karate	0.62	0.65	0.68	0.60	0.65	0.69	0.74
Dolphins	0.42	0.59	0.59	0.42	0.56	0.59	0.65
Football	0.74	0.70	0.75	0.71	0.73	0.76	0.80
Amazon	0.60	0.64	0.65	0.50	0.70	0.57	0.82
DBLP	0.80	0.83	0.85	0.81	0.84	0.86	0.91
Ego-Facebook	0.60	0.52	0.58	0.54	0.68	0.66	0.72

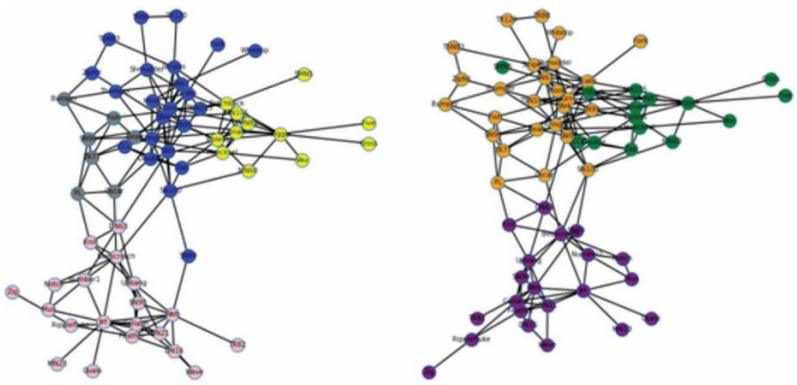


Figure 8: Dolphin Community structure (Left) Ground Truth (Right) Community Structure detected by Hybrid Model

Table 6: Final ranking for the compared algorithms in terms of NMI on six Datasets

Networks	Louvain [37]	Spinglass [76]	Walktrap [77]	FLPA [22]	Girvan Newman [40]	Infomap [78]	Proposed
Karate	6	8	5	11	10	8	1
Dolphins	3	12	8	4	7	5	1
Football	7	3	2	8	5	6	1
Amazon	3	5	7	4	6	4	2
DBLP	10	11	6	5	5	7	1
Ego-Facebook	11	13	8	4	7	9	5

This is further corroborated by the NMI metrics in Table 6, demonstrating how the suggested algorithm performs better than alternative methods in obtaining higher values. This figure 7 demonstrates that the suggested method works better for the Karate network, despite having a lower value than the Walktrap [77], FLPA [22], Louvain [37], and Spinglass [76] algorithms. Regardless of the size of the dataset, our method computes a community structure that is more congruent with the ground truth than opposing approaches. As Table 6 makes clear, our idea has further important advantages. The stability of our proposed hybrid model algorithm surpasses that of other algorithms, including Louvain [37], FLPA [22], and Spinglass [76]. This is true as our concept does not rely on a haphazard process.

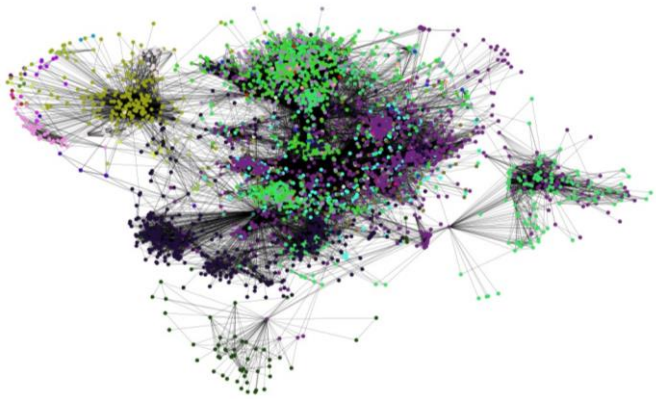


Figure 9: Clustering of different Communities

Additionally, our suggested hybrid model works well in dense graphs, providing significant NMI values, which is seen to be an essential benefit for social network community finding. Figure 9 displays the community structure that the hybrid model identified. To summarize, as discussed in [19] the quick identification of a community structure that closely mimics the actual structure. Therefore, our approach performs exceptionally well in locating significant communities inside social networks.

To demonstrate how much better the recommended method is, the experiment selects the baseline approaches, which include several traditional community detection techniques and well-known clustering techniques. To assess the performance of the Proposed technique, it will be compared to existing embedding-based baselines with existing graph embedding baseline approaches [1–3] for CD, we fit networks into them. After that, we extract community divisions from the lowdimensional vector space we have learned using the Algorithm 5. Using the data from Tables 4 and 5, we compare the results with our proposed technique and six existing community detection methods on the network of real-world ground truth datasets. We used the modularity value (Q) and the NMI value in the evaluation procedure following [6, 8, 10]. The strategy achieves all six of the highest NMI values and all five of the highest Q values in the real-world community datasets. Even if there is a tiny discrepancy between the Q values produced from the Ego-Facebook dataset and the Q values in the Dolphins dataset, the largest NMI value of 1 is attained, demonstrating that it is exactly the correct community for the actual categorization. In the DBLP dataset, the recommended approach performs better than the other methods [4, 5]and yields more accurate information, obtaining the greatest NMI value of 0.91. The Amazon dataset has higher Q and NMI values, and the NMI is closer to 1, which is more consistent with data from genuine communities. With the greatest outcomes and the greatest rationality and effectiveness compared to the other algorithms, which generated the least modularity, was DBLP.

The empirical results give compelling proof that the suggested hybrid algorithm performs well at finding meaningful communities across a wide range of real-world networks. The approach got the best modularity (Q = 0.86) and NMI (NMI = 0.91) on the DBLP dataset, beating the best methods like Louvain, Infomap, and Walktrap. On networks like Karate and Dolphins, the model also consistently came in first in modularity and NMI, which means it did a superior job of grouping nodes by structure. The results show that combining eigenvector and proximity centrality with embedding methods makes it easier to find important nodes and community boundaries. The higher modularity shows that the algorithm is good at building communities that are dense on the inside and sparse on the outside. The high NMI scores show that the results match the ground truth. So, the suggested model shows that it can be used in a wide range of situations, is accurate, and is useful in real life with complicated graph topologies.

6.4. Time Complexity

The greedy method requires 0(n+m) for each iteration. CN iterations were required for this method to function. Consequently, the overall time complexity of this algorithm was 0(m + n)(n). Since there are more edges than nodes in the graph, 0(mn) has the highest level of complexity. In the end, the temporal complexity of the method can be represented by Equation 18 as follows:

$$T(N) = (m.d_w^2) + O(n^2) + O(ns.nw) \tag{18}$$

which is almost equivalent to O ($m.d^2$).

7. Conclusion

Easy communication between individuals on a single platform has been made possible by the growth of the web and the development of SNS. A graph containing nodes and edges linking the nodes can be used to depict a social network. The edges show how these entities interact with one another, whereas the nodes represent the individuals or entities. Individuals with comparable choices, tastes, and preferences who frequently connect on social media platforms create virtual groups or communities. It entails recognizing cohesive groups with related entities and setting cohesive groups apart from other groupings. Numerous approaches have been put out for community detection, each taking a distinct angle on the issue. Large-

scale graph handling community detection techniques, however, are now required since complex and huge networks are emerging across multiple sectors. This research suggests a new method to detect social networks of people depending on community knowledge and embedding spaces of similar nodes. To improve CD in social networks, we integrate eigenvector centrality and closeness measurements. Comprehensive tests on real-world networks show our proposal's effectiveness. The experimental findings demonstrate how robust and efficient the suggested method is, and how well it performs in large-scale graphs when compared to other well-known algorithms. The complexity of the algorithm stays quadratic about the number of vertices and linear about the number of iterations. This algorithm is still not very good at solving big data challenges. However, there is still research to see if changing the data format will bring the complexity down to almost a linear level. It is also possible to develop better disassembly methods for communities and nodes, which would require fewer iterations to achieve increased modularity. This method can be applied to real-world problems to detect communities in the bio-informatics industry, which entails assembling the major proteins involved in cancer and searching for functional connections within the resulting communities.

7.1. Limitations

The quality of the user interaction data may affect the computed centrality measures. Inaccurate or lacking data may lead to incorrect inferences. Online sites may restrict access to user interaction data due to privacy limitations. It is imperative to consider ethical considerations when collecting and utilizing this type of data. Computing proximity centrality and eigenvector centrality can be computationally costly for very large user interaction networks. This might limit how large the method can be scaled for real-world applications. When paired with other user data, careful feature engineering can be required to guarantee that centrality ratings make a substantial contribution to the clustering process. User interaction networks are dynamic, and communities are subject to change over time. It's possible that the communities that have been identified don't precisely match the most recent configuration of the network.

Funding Statement: This research did not receive any specific grant from any funding agency.

Conflicts of Interest: There are no financial, personal, or professional conflicts of interest.

Data Availability: This study utilizes six real-world datasets, which were obtained from publicly accessible repositories.

References

[1] Bolorunduro, Janet Oluwasola, and Zhaonian Zou. "Community detection on multi-layer graph using intra-layer and inter-layer linkage graphs (cdmiilg)." *Expert Systems with Applications* 238 (2024): 121713.

[2] Yang, Cheng, Jixi Liu, Yunhe Yan, and Chuan Shi. "Fairsin: Achieving fairness in graph neural networks through sensitive information neutralization." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 8, pp. 9241-9249. 2024.

[3] Gadár, László, and János Abonyi. "Finding multifaceted communities in multiplex networks." *Scientific Reports* 14, no. 1 (2024): 14521.

[4] Cavoretto, Roberto, Alessandra De Rossi, Sandro Lancellotti, and Federico Romaniello. "Node-bound communities for partition of unity interpolation on graphs." *Applied Mathematics and Computation* 467 (2024): 128502.

[5] Christensen, Alexander P., Luis Eduardo Garrido, Kiero Guerra-Peña, and Hudson Golino. "Comparing community detection algorithms in psychometric networks: A Monte Carlo simulation." *Behavior Research Methods* 56, no. 3 (2024): 1485-1505.

[6] Xia, Yuanxing, Qingshan Xu, Jicheng Fang, Rongchuan Tang, and Pengwei Du. "Bipartite graph-based community-to-community matching in local energy market considering socially networked prosumers." *Applied energy* 353 (2024): 122245.

[7] Baruah, Bikash, Manash P. Dutta, Subhasish Banerjee, and Dhruba K. Bhattacharyya. "A novel density based community detection algorithm and its application in detecting potential biomarkers of ESCC." *Journal of Computational Science* 81 (2024): 102344.

[8] Zheng, Jianxing, Suge Wang, Deyu Li, and Bofeng Zhang. "Personalized recommendation based on hierarchical interest overlapping community." *Information Sciences* 479 (2019): 55-75.

[9] Adraoui, Meriem, Asmaâ Retbi, Mohammed Khalidi Idrissi, and Samir Bennani. "Maximal cliques based method for detecting and evaluating learning communities in social networks." *Future Generation Computer Systems* 126 (2022): 1-14.

[10] Fortunato, Santo. "Community detection in graphs." *Physics reports* 486, no. 3-5 (2010): 75-174.

[11] Yilmaz, L. Safak, and Albertha JM Walhout. "Metabolic network modeling with model organisms." *Current opinion in chemical biology* 36 (2017): 32-39.

[12] Newman, Mark EJ, and Michelle Girvan. "Finding and evaluating community structure in networks." *Physical review E* 69, no. 2 (2004): 026113.

[13] Ahajjam, Sara, Mohamed El Haddad, and Hassan Badir. "A new scalable leader-community detection approach for community detection in social networks." *Social Networks* 54 (2018): 41-49.

[14] Azaouzi, Mehdi, Delel Rhouma, and Lotfi Ben Romdhane. "Community detection in large-scale social networks: state-of-the-art and future directions." *Social Network Analysis and Mining* 9, no. 1 (2019): 23.

[15] Rostami, M., Farrahi, V., Ahmadian, S., Jalali, S.M.J. and Oussalah, M., 2023. A novel healthy and time-aware food recommender system using attributed community detection. *Expert Systems with Applications*, *221*, p.119719.

[16] Rostami, Mehrdad, Mourad Oussalah, Kamal Berahmand, and Vahid Farrahi. "Community detection algorithms in healthcare applications: a systematic review." *IEEE Access* 11 (2023): 30247-30272.

[17] Rostami, Mehrdad, Usman Muhammad, Saman Forouzandeh, Kamal Berahmand, Vahid Farrahi, and Mourad Oussalah. "An effective explainable food recommendation using deep image clustering and community detection." *Intelligent Systems with Applications* 16 (2022): 200157.

[18] Alotaibi, Norah, and Delel Rhouma. "A review on community structures detection in time evolving social networks." *Journal of King Saud University-Computer and Information Sciences* 34, no. 8 (2022): 5646-5662.

[19] Xu, Mengjia. "Understanding graph embedding methods and their applications." *SIAM Review* 63, no. 4 (2021): 825-853.

[20] Kumar, Prashant, Raghav Jain, Shivam Chaudhary, and Sanjay Kumar. "Solving community detection in social networks: A comprehensive study." In *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 239-345. IEEE, 2021.

[21] Bhattacharya, Riju, Naresh Kumar Nagwani, and Sarsij Tripathi. "A community detection model using node embedding approach and graph convolutional network with clustering technique." *Decision Analytics Journal* 9 (2023): 100362.

[22] Traag, Vincent A., and Lovro Šubelj. "Large network community detection by fast label propagation." *Scientific Reports* 13, no. 1 (2023): 2701.

[23] Chen, Gaolin, and Shuming Zhou. "A novel overlapping community detection strategy based on Core-Bridge seeds." *International journal of machine learning and cybernetics* 15, no. 6 (2024): 2131-2147.

[24] Mohamed, El-Moussaoui, Tarik Agouti, Abdessadek Tikniouine, and Mohamed El Adnani. "A comprehensive literature review on community detection: Approaches and applications." *Procedia Computer Science* 151 (2019): 295-302.

[25] Mothe, Josiane, Karen Mkhitaryan, and Mariam Haroutunian. "Community detection: Comparison of state of the art algorithms." In *2017 Computer Science and Information Technologies (CSIT)*, pp. 125-129. IEEE, 2017.

[26] Newman, Mark EJ. "Fast algorithm for detecting community structure in networks." *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics* 69, no. 6 (2004): 066133.

[27] Chintalapudi, S. Rao, and MHM Krishna Prasad. "Finding research groups using modularity based community detection algorithm." In *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pp. 65-69. IEEE, 2016.

[28] Behera, Ranjan Kumar, S. K. Rath, and Monalisa Jena. "Spanning tree based community detection using min-max modularity." *Procedia Computer Science* 93 (2016): 1070-1076.

[29] Krishnan, S. Gokula, S. Karthika, and S. Bose. "Detection of communities in dynamic social networks." In *2016 International Conference on Recent Trends in Information Technology (ICRTIT)*, pp. 1-6. IEEE, 2016.

[30] Jiang, Tianhua, Chao Zhang, Huiqi Zhu, and Guanlong Deng. "Energy-efficient scheduling for a job shop using grey wolf optimization algorithm with double-searching mode." *Mathematical Problems in Engineering* 2018, no. 1 (2018): 8574892.

[31] Gutiérrez, Inmaculada, Daniel Gómez, Javier Castro, and Rosa Espínola. "A new community detection problem based on bipolar fuzzy measures." In *Computational Intelligence and Mathematics for Tackling Complex Problems 2*, pp. 91-99. Cham: Springer International Publishing, 2022.

[32] Huang, Mingqing, Qingshan Jiang, Qiang Qu, and Abdur Rasool. "An overlapping community detection approach in ego-splitting networks using symmetric nonnegative matrix factorization." *Symmetry* 13, no. 5 (2021): 869.

[33] Zhu, Jiajing, Yongguo Liu, Hao Wu, Zhi Chen, Yun Zhang, Shangming Yang, Changhong Yang, Wen Yang, and Xindong Wu. "A no self-edge stochastic block model and a heuristic algorithm for balanced anti-community detection in networks." *Information Sciences* 518 (2020): 95-112.

[34] Yin, Ying, Yuhai Zhao, He Li, and Xiangjun Dong. "Multi-objective evolutionary clustering for large-scale dynamic community detection." *Information Sciences* 549 (2021): 269-287.

[35] Tabarzad, Mohammad Ali, and Ali Hamzeh. "A heuristic local community detection method (HLCD)." *Applied Intelligence* 46, no. 1 (2017): 62-78.

[36] Zhang, Xingyi, Congtao Wang, Yansen Su, Linqiang Pan, and Hai-Feng Zhang. "A fast overlapping community detection algorithm based on weak cliques for large-scale networks." *IEEE Transactions on Computational Social Systems* 4, no. 4 (2017): 218-230.

[37] Zhou, Xiaojun, Ke Yang, Yongfang Xie, Chunhua Yang, and Tingwen Huang. "A novel modularity-based discrete state transition algorithm for community detection in networks." *Neurocomputing* 334 (2019): 89-99.

[38] Blondel, Vincent D., Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. "Fast unfolding of communities in large networks." *Journal of statistical mechanics: theory and experiment* 2008, no. 10 (2008): P10008.

[39] Traag, Vincent A., Ludo Waltman, and Nees Jan Van Eck. "From Louvain to Leiden: guaranteeing well-connected communities." *Scientific reports* 9, no. 1 (2019): 1-12.

[40] Girvan, Michelle, and Mark EJ Newman. "Community structure in social and biological networks." *Proceedings of the national academy of sciences* 99, no. 12 (2002): 7821-7826.

[41] Ghoshal, Arnab Kumar, Nabanita Das, Subhasis Bhattacharjee, and Goutam Chakraborty. "A Fast Parallel Genetic Algorithm Based Approach for Community Detection in Large Networks." In *COMSNETS*, pp. 95-101. 2019.

[42] Alhijawi, Bushra, and Arafat Awajan. "Genetic algorithms: Theory, genetic operators, solutions, and applications." *Evolutionary Intelligence* 17, no. 3 (2024): 1245-1256.

[43] Zeng, Xiangxiang, Wen Wang, Cong Chen, and Gary G. Yen. "A consensus community-based particle swarm optimization for dynamic community detection." *IEEE transactions on cybernetics* 50, no. 6 (2019): 2502-2513.

[44] Hafez, Ahmed Ibrahem, Aboul ella Hassanien, Aly A. Fahmy, and Mohamed Fahmy Tolba. "Community detection in social networks by using Bayesian network and Expectation Maximization technique." In *13th International Conference on Hybrid Intelligent Systems (HIS 2013)*, pp. 209-214. IEEE, 2013.

[45] Srinivas, Sharan, and Chandrasekharan Rajendran. "Community detection and influential node identification in complex networks using mathematical programming." *Expert Systems with Applications* 135 (2019): 296-312.

[46] Cheng, Jianjun, Xing Su, Haijuan Yang, Longjie Li, Jingming Zhang, Shiyan Zhao, and Xiaoyun Chen. "Neighbor similarity based agglomerative method for community detection in networks." *Complexity* 2019, no. 1 (2019): 8292485.

[47] You, Xuemei, Yinghong Ma, and Zhiyuan Liu. "A three-stage algorithm on community detection in social networks." *Knowledge-Based Systems* 187 (2020): 104822.

[48] Kasoro, Nathanaël, Selain Kasereka, Elie Mayogha, Ho Tuong Vinh, and Joël Kinganga. "PercoMCV: A hybrid approach of community detection in social networks." *Procedia Computer Science* 151 (2019): 45-52.

[49] Tahir, Noman, Ali Hassan, Muhammad Asif, and Shahbaz Ahmad. "MCD: mutually connected community detection using clustering coefficient approach in social networks." In *2019 2nd International Conference on Communication, Computing and Digital systems (C-CODE)*, pp. 160-165. IEEE, 2019.

[50] Bai, Liang, Jiye Liang, Hangyuan Du, and Yike Guo. "A novel community detection algorithm based on simplification of complex networks." *Knowledge-Based Systems* 143 (2018): 58-64.

[51] Rustamaji, Heru C., Yustina S. Suharini, Angga A. Permana, Wisnu A. Kusuma, Sri Nurdiati, Irmanida Batubara, and Taufik Djatna. "A network analysis to identify lung cancer comorbid diseases." *Applied Network Science* 7, no. 1 (2022): 30.

[52] Fortunato, Santo, and Marc Barthelemy. "Resolution limit in community detection." *Proceedings of the national academy of sciences* 104, no. 1 (2007): 36-41.

[53] Guo, Jiahao, Pramesh Singh, and Kevin E. Bassler. "Resolution limit revisited: community detection using generalized modularity density." *Journal of Physics: Complexity* 4, no. 2 (2023): 025001.

[54] Miyauchi, Atsushi, and Yasushi Kawase. "Z-score-based modularity for community detection in networks." *PloS one* 11, no. 1 (2016): e0147805.

[55] Fortunato, Santo, and Darko Hric. "Community detection in networks: A user guide." *Physics reports* 659 (2016): 1-44.

[56] He, Kun, Yingru Li, Sucheta Soundarajan, and John E. Hopcroft. "Hidden community detection in social networks." *Information Sciences* 425 (2018): 92-106.

[57] Chattopadhyay, Swarup, and Debasis Ganguly. "Node2vec with weak supervision on community structures." *Pattern Recognition Letters* 150 (2021): 147-154.

[58] Orman, Günce Keziban, Vincent Labatut, and Hocine Cherifi. "Qualitative comparison of community detection algorithms." In *International conference on digital information and communication technology and its applications*, pp. 265-279. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[59] Freeman, Linton C. "A set of measures of centrality based on betweenness." *Sociometry* (1977): 35-41.

[60] Freeman, Linton C. "Centrality in social networks conceptual clarification." *Social networks* 1, no. 3 (1978): 215-239.

[61] Borgatti, Stephen P., and Martin G. Everett. "A graph-theoretic perspective on centrality." *Social networks* 28, no. 4 (2006): 466-484.

[62] Bavelas, Alex. "Communication patterns in task-oriented groups." *Journal of the acoustical society of America* (1950).

[63] Spizzirri, Leo. "Justification and application of eigenvector centrality." *Algebra in Geography: Eigenvectors of Network* (2011).

[64] Ding, De-wu, and Xiao-qing He. "Notice of Retraction: Application of eigenvector centrality in metabolic networks." In *2010 2nd International Conference on Computer Engineering and Technology*, vol. 1, pp. V1-89. IEEE, 2010.

[65] Shiokawa, Hiroaki, Yasuhiro Fujiwara, and Makoto Onizuka. "Scan++ efficient algorithm for finding clusters, hubs and outliers on large-scale graphs." *Proceedings of the VLDB Endowment* 8, no. 11 (2015): 1178-1189.

[66] Gmati, Haifa, Amira Mouakher, Antonio Gonzalez-Pardo, and David Camacho. "A new algorithm for communities detection in social networks with node attributes." *Journal of Ambient Intelligence and Humanized Computing* 15, no. 2 (2024): 1779-1791.

[67] Guerreiro, Lucas, Filipi Nascimento Silva, and Diego Raphael Amancio. "Identifying the perceived local properties of networks reconstructed from biased random walks." *Plos one* 19, no. 1 (2024): e0296088.

[68] Hamedani, Masoud Reyhani, and Sang-Wook Kim. "JacSim: An accurate and efficient link-based similarity measure in graphs." *Information Sciences* 414 (2017): 203-224.

[69] Newman, Mark EJ. "Modularity and community structure in networks." *Proceedings of the national academy of sciences* 103, no. 23 (2006): 8577-8582.

[70] Csardi, Gabor, and Tamas Nepusz. "The igraph software." *Complex syst* 1695 (2006): 1-9.

[71] Hagberg, Aric, Pieter J. Swart, and Daniel A. Schult. *Exploring network structure, dynamics, and function using NetworkX*. No. LA-UR-08-05495; LA-UR-08-5495. Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

[72] Avrachenkov, Konstantin E., Aleksei Yu Kondratev, and Vladimir V. Mazalov. "Cooperative game theory approaches for network partitioning." In *International Computing and Combinatorics Conference*, pp. 591-602. Cham: Springer International Publishing, 2017.

[73] Lusseau, David, Karsten Schneider, Oliver J. Boisseau, Patti Haase, Elisabeth Slooten, and Steve M. Dawson. "The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations: can geographic isolation explain this unique trait?." *Behavioral ecology and sociobiology* 54, no. 4 (2003): 396-405.

[74] Yang, Jaewon, and Jure Leskovec. "Defining and evaluating network communities based on ground-truth." In *Proceedings of the ACM SIGKDD workshop on mining data semantics*, pp. 1-8. 2012.

[75] Jure, Leskovec. "Snap datasets: Stanford large network dataset collection." *Retrieved December 2021 from http://snap. stanford. edu/data* (2014).

[76] Eaton, Eric, and Rachael Mansbach. "A spin-glass model for semi-supervised community detection." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 26, no. 1, pp. 900-906. 2012.

[77] Smith, Natalie R., Paul N. Zivich, Leah M. Frerichs, James Moody, and Allison E. Aiello. "A guide for choosing community detection algorithms in social network studies: The question alignment approach." *American journal of preventive medicine* 59, no. 4 (2020): 597-605.

[78] Zeng, Jianping, and Hongfeng Yu. "A distributed infomap algorithm for scalable and high-quality community detection." In *Proceedings of the 47th International Conference on Parallel Processing*, pp. 1-11. 2018.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Predicting Colorectal Cancer Using Machine Learning and Worldwide Dietary Data

Muhammad Sanaullah[1, *], Muhammad Kashif[1]

[1]Department of Computer Science, Bahauddin Zakariya University, Multan, 60800, Pakistan

[*]Corresponding Author: Muhammad Sanaullah. Email: muhammad.sanaullah@aumc.edu.pk

Received: 29 November 2024; Revised: 08 January 2025; Accepted: 10 February 2025; Published: 20 March 2025

AID: 004-01-000048

Abstract: Colorectal Cancer (CRC) is considered to be a substantial catastrophic disease and the third most commonly reported type of cancer worldwide. By performing proactive screening of patients for CRC detection, it has been found that its is most prominently diagnosed in younger adults. However, most of the recently published papers have primarily focused upon the implication of statistical machine learning algorithms for CRC diagnosis in older adults with the aid of small-scale datasets, which are unable to depict acceptable performance in practice for large populations. So, it is crucial to assess machine learning algorithms on big datasets from varied areas and socio demographics, including both younger and older persons. The Centre for Disease Control and Prevention acquired a dataset of 109,343 individuals from colorectal cancer research in South Korea, India, Canada Mexico, Italy, Sweden, and the US. This worldwide dietary database was supplemented using publicly available information from several sources. In this study, we have evaluated performance of nine supervised and unsupervised machine learning methods on the aggregated dataset. Both type of tested models (i.e., supervised and unsupervised) models accurately predicted CRC and non-CRC traits. Among the nine tested models, artificial neural network (ANN) has achieved best performance, while attaining a misclassification rate of 1% and 3% for CRC and non-CRC respectively. ANN model has depicted extraordinary performance over diverse datasets, which make it a suitable choice for CRC diagnosis in both young and elderly persons. Using optimum algorithms and ensuring high screening compliance can significantly enhance early cancer detection and increase the success rate of prompt treatments.

Keywords: Colonel Cancer; Machine Learning; Cancer Screening; Early Cancer Detection;

1. Introduction

The recent revolution of artificial intelligence (AI) in 21st century has opened up new opportunities to enhance healthcare services by introducing advanced healthcare data analytics solutions, while overcoming conventional statistical and research constraints [1, 2]. Colorectal cancer (CRC) is one of the problems facing healthcare today. After lung and breast cancers, colorectal cancer (CRC) is the second most prevalent cause of cancer-related death globally and the third most often diagnosed disease [3, 4]. An estimated 1.93 million new cases of colorectal cancer were detected in 2020, representing 10% of all cancer cases worldwide [5]. Effective population-wide screening and monitoring initiatives that have been rapidly and proactively implemented may be responsible for the rising number of CRC cases worldwide [6, 7].

Nonetheless, CRC death rate remains high, with as estimated 0.94 million mortalities documented for this disease in 2020, accounting for 9.4% of all cancer deaths worldwide [5]. These statistics highlight the need of active health screening for the prevention of CRC in younger generation (under 50 years) specifically due to the high reporting rate of early-onset cases in technologically advanced countries, while an elevated observed rate in CRC incidence detection in developing and emerging economies [8, 9]. The medical advancements to enhance treatment options for CRC, such as by performing surgical and endoscopy-based interventions, targeted chemotherapy, immunotherapy and radiotherapy have led to improved survival rates and quality of life [9, 10]. Towards this end, Saudi Ministry of Health (MoH) has also taken a prominent measure i.e., by recommending early and periodic screening for CRC primarily on the basis of patient's history and symptoms. The two primary groups who are primarily focused for the early diagnosis of this catastrophic disease are individual having low risk i.e., between age 45 to 75, and secondly individuals with high risk of CRC i.e., who may have a family history of cancer or gets exposed to radiation therapy in their childhood. The colorectal examination is divided into two types: 1) the fecal occult blood test (FOBT) also known as fecal immunological test (FIT) and secondly 2) the whole colonoscopy [11]. Early detection of CRC leads to a better prognosis for treatment, but it still poses significant public health and financial issues (9). In 2015, the economic impact of CRC in Europe was projected to be 19 billion euros, including hospital bills, lost productivity, premature mortality, and informal care costs [11]. Early-onset CRC pathological characteristics are sporadic and require further investigation to properly understand the underlying processes and risk factors [9]. With the advancement of digital technologies, health information systems can efficiently collect high-quality CRC data from a larger patient population. This has allowed data science to provide a new route for increasing understanding about CRC through research and development.

Machine-learning methods have successfully predicted CRC based on genomic data, indicating inherited propensity in some situations [12, 13]. However, genetic disorders are persistent and unchangeable risk factors. Dietary restriction is a highly effective way to prevent CRC, as it is linked to a lifestyle associated with globalization [4, 14, 15-16]. As the food business and supply chain become more globalized, it's crucial to do data science study on how global diets impact CRC prediction.

This research aims to develop ML models to identify key dietary components influencing CRC risk. By utilizing publicly available global dietary data, we seek to improve understanding of how dietary habits contribute to CRC and enhance predictive analytics for early detection and prevention strategies. In this research, we used exploratory unsupervised and supervised machine learning-based models to examine the key dietary components in predicting CRC labelling.

2. Problem Statement

CRC is the second leading cause of cancer-related deaths worldwide. Despite improvements in early detection methods, such as fecal tests and colonoscopy, and improvements in treatment options, CRC death count is high. Genetic predisposition is a major risk factor, but lifestyle and dietary habits, influenced by globalization, also play a crucial role in CRC development. The existing studies primarily focus on genomic data, overlooking the impact of dietary patterns on CRC risk.

3. Related Work

The authors of [17] employed feature selection methods and machine learning algorithms to identify colon cancer. For feature selection, the maximum degree greedy (MDG) and malondialdehyde (MDA) algorithms were employed. AdaBoost, logistic regression (LR), KNN, SVM, and RF Algorithms have been used on a public dataset consisting of 2000 genes and 62 instances. Among them are 22 normal patients and 40 abnormal patients. The outcome demonstrated that, the random forest classifier together with a features selection approach has achieved the best accuracy of 95.2%. Only genes are used as features in the model.

The study in [18] uses an ensemble classifier approach to divide tissues into normal and pathological categories. Filtering and wrapping are the feature selection techniques that were applied. At Pablo de

Olavide University's Bioinformatics Research Group, 62 patients and 1200 gene expressions were subjected to the machine learning algorithms KSVM, RF, eXtreme ensemble, KNN and Gradient Boosting (XGB). Among them are 22 normal patients and 40 abnormal patients. This model's results showed that the proposed ensemble learning based approach has achieved the best accuracy of 91.7%.

The authors of [19] examined the challenge of identifying colorectal cancer. The primary model that has been employed in this study is modified Harmony Search Algorithm (Z-FS-KM-MHS). The Princeton University Gene Expression Project provided 2000 genes in all. Z-FS-KM-MHS obtained an accuracy of up to 94.4%, according to the results. Many genes were employed in the model. In contrast to other research, this approach can be used to study genetically based disorders like breast cancer.

Hamida et al. has presented a deep Convolutional Neural Network (CNN) model for dividing colon pictures into normal and nonnormal categories. In Germany, the UNET and SEGNET models were the primary models used for 100,000 histopathology scans [46–50]. SEGNET achieved 99.5% high-performance accuracy. The scientists came to the conclusion that DL performs better at picture classification than ML when dealing with large-scale photos. In contrast to, colon cancer was classified using pictures [20].

The authors of [21] enhanced the colon cancer diagnosis. To do so, selected machine learning models have been trained over two publicly available datasets, which primarily incorporate 98 samples and 9457 genes. The ML models that have been selected and trained in this study are decision trees (DT), naive Bayes (NB), Support Vector Machine (SVM) and KNN. The results of the study reveal that the KNN and DT models have achieved best classification performance over the first selected dataset, while NB model has attained best results on second dataset.

The issue of colonoscopy failure to detect polyps is examined by the authors in [22]. The primary technique used was DL on 27,113 colonoscopy pictures and 1290 patients which belongs to Sichuan Provincial People's Hospital's Endoscopy Center. The employed DL classifier has achieved a Per-image detection rate of 91.6%. But the algorithm just finds polyps. In order to identify colon tumors from biopsy data, another prominent methodology has been presented by author of [23], who basically employed the Density-Based Spatial Clustering of Applications with Noise (DB-SCAN) algorithm, which classifies healthy cells from dangerous ones. One hundred photos gathered from Zendo repositories were used to test the algorithm. According to the findings, the model detected colon cancers with 99% accuracy score, 85.4% sensitivity score, and 87.6% specificity score [23].

Jørgensen et al. extracted information from cell nuclei to determine whether the tissue was malignant or benign. To do so, author has exploited a multi-classifier based approach, which include RF, k-means clustering, color deconvolution, local adaptive thresholding, and separation of cell within ROI on 87 distinct colon tissue slides. Consequently, the proposed algorithm's sensitivity, specificity, accuracy, and area under the curve (AUC) were 0.96, 0.88, 0.92, and 0.91, respectively [24].

Using ANNs and a feature selection approach, authors in [25] has presented a model for the classification of lung and colon cancer. The creators of this model used a publicly available dataset with 62 cases and 2000 genes. For the two classifications of cancer and normal, the classification accuracy was 98.4%. Furthermore, the authors discovered that the feature selection approach might improve the model's classification accuracy.

In his study, Choi et al has presented a deep learning-based computer-aided diagnosis (CAD) system for the multi-classification of pathologic histology of colorectal adenoma in four different classes. The model developed a diagnostic method that primarily forecast tissue adenoma of the colon and rectum by using CNN's algorithm, while employing 3400 computed tomography (CT) images gained from a Korea based Hospital (KUMC) and a CAD. The authors then contrasted the system's output with the experts' findings. According to the findings, the classification had a sensitivity of 77.25% and a specificity of 92.42%. Furthermore, it closely matched the experts' findings. Lack of sufficient samples to evaluate the model's validity is one of the authors' challenges [26].

Using a self-speed transmission network, Yao et al. suggested automatically classifying and segmenting colorectal images into three groups: cancers, polyps, and normal tissue. To enhance the outcome, a pre-

trained ImageNet network was first used, and then 3061 photos were used. The trained STVGG network was employed for further analysis for colon rectal classification after the Unet network architecture was utilized for segmentation. The model has achieved good segmentation and classification accuracy. The combination of two goals—classification and segmentation—sets this paper apart from the other research. Additionally, it employed self-learning to learn the challenging sample, address the imbalance issue, and improve performance [27].

Table 1: Literature Review Analysis

Ref.	Dataset Instances	ML Model	Achieved Accuracy
[17]	62 patients and 1200 gene expression	AdaBoost, KNN, logistic regression (LR), SVM, and RF	95.161%
[18]	62 patients and 1200 gene expression	RF, KSVM, eXtreme Gradient Boosting (XGB), KNN,	91.67%.
[19]	2000 genes	Z-FS-KM-MHS	94.36%
[20]	100,000 histopathology scans	CNN, SEGNET	99.5%
[21]	9457 genes and 98 samples	SVM, Naïve Bayes, Decision Tree, KNN	-
[23]	27,113 colonoscopy pictures and 1290 patients	DB-SCAN	99%
[24]	87 colon tissue	RF, K mean Clustering	92%
[25]	62	ANN	98.4%.
[26]	3400 computed tomography (CT) images	CNN	sensitivity = 77.25%, specificity = 92.42%.
[27]	3061 photos	Self-paced Transfer VGG (STVGG)	-

The comparison analysis of literature is given in table 1 where 1st column represents the references of selected paper for analysis, 2nd column represents the dataset used in study, 3rd column represents the ML model used and last column represents the highest achieved accuracy. The datasets used in the studies range from small-scale gene expression data with a few dozen patients [17, 18], [25] to large-scale histopathology and colonoscopy image datasets containing thousands of samples [20], [23], [26]. The nature of the datasets significantly influences the choice of ML models. Gene expression-based studies [17-19], [21], [24] used traditional ML models such as 1) RF, 2) KNN, 3) SVM and 4) eXtreme Gradient Boosting (XGB), which are effective for structured data analysis. On the other hand, image-based datasets in studies [20], [23], [26] based on deep learning models like Convolutional Neural Networks (CNN) and SEGNET, which are well-suited for feature extraction from complex medical images.

Performance comparisons indicate that deep learning models perform exceptionally well on medical imaging tasks. CNN and SEGNET achieved the highest accuracy of 99.5% in histopathology image classification [20], while DB-SCAN [23] achieved 99% accuracy in colonoscopy image analysis. In traditional ML approaches, AdaBoost, KNN, Logistic Regression (LR), SVM, and RF [17] reached 95.161% accuracy, demonstrating the strength of ensemble methods in gene expression data analysis. Additionally, Artificial Neural Networks (ANN) [25] achieved an impressive 98.4% accuracy, indicating that neural networks remain competitive in structured data classification. Moderate performance was observed in RF and clustering-based approaches [24], which achieved 92% accuracy, and Z-FS-KM-MHS [19] with 94.36% accuracy. Notably, study [26] reported sensitivity (77.25%) and specificity (92.42%) instead of overall accuracy, offering a different perspective on model effectiveness. However, some studies [21], [27] did not report accuracy, limiting direct comparisons.

Overall, the findings indicate that deep learning models, particularly CNN-based architectures, dominate medical image classification, while traditional ML methods such as SVM, RF, and XGB continue to excel in gene expression data analysis. Hybrid and clustering-based approaches, such as DB-SCAN [23] and self-paced transfer learning [27], show potential in specific applications. Future research should focus on integrating multimodal data (gene expression and imaging) to improve predictive accuracy while enhancing the interpretability of AI-driven healthcare solutions.

4. Methodology

The proposed methodology composed of Dataset selection, Data Preprocessing, Feature Selection, Data Normalization and Classification as shown in Fig. 1.

4.1. Dataset

The Centers for Disease Control and Prevention also renowned as the Global Dietary database, and publicly available institutional websites provided the dietary-related colorectal cancer data [28-34]. Canada, Korea, Argentina, Ecuador, Bangladesh, Estonia, Bulgaria, Finland, China, India, Ethiopia, Israel, Germany, Malaysia, Iran, Kenya, Mexico, Portugal, United States, the Philippines, Japan, Mozambique, Italy, Sweden and Tanzania were among the 25 nations that made up the original combined data. These data sets were collected using comparable techniques, such as cross-sectional surveys and food questionnaires. Following that, the various data sets were combined and extrapolated using the same dietary features. Excluded were characteristics that did not appear in all of the data sets.

4.2. Data Preprocessing

Only English-language data sets are included in this analysis. For the purpose of standardization, features with disparate measurement units were transformed. A process of cleaning was used, such as listwise elimination of features having greater than 50% missing values, duplicate characteristics, and ineligible situations. There were still 3,520,586 valid data points at this point.

During preprocessing a specific sample of dataset containing 109,342 cases have been extracted to tackle the computational cost issue. This data sampling has been done through a multi-stage, proportionate random sample method. Of these, 7,326 (6.7%) cases had positive colorectal cancer labels, which are derived for seven distinct countries worldwide.

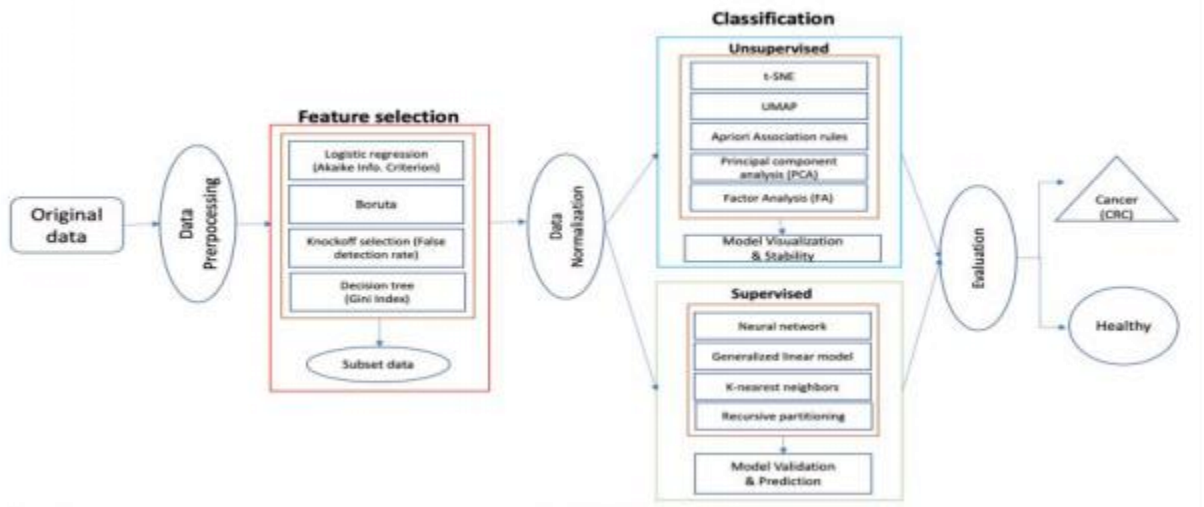


Figure 1: Proposed Methodology

4.3. Feature Selection

A two-phase feature selection process was used. Three distinct processes are involved in step one: Boruta, Knock of selection, and logistic regression (LR). Each index was filtered out using LR in order to decrease redundant features by utilizing the step AIC function in the MOSS package to calculate a stepwise iterative process of forward addition and backward removal. The primary aim of forward addition is to add significant features to a null set of features, while the aim of backward removal is to remove the worst-performing features from the list of full features [35]. Based on statistical analysis, the most statistically significant features are considered to be most economical model with the lowest Akaike Information Criterion (AIC) were used to choose variables. Then, a randomized wrapper technique called Boruta was used, which gradually eliminates characteristics that are more important and statistically insignificant than those of random probes [36].

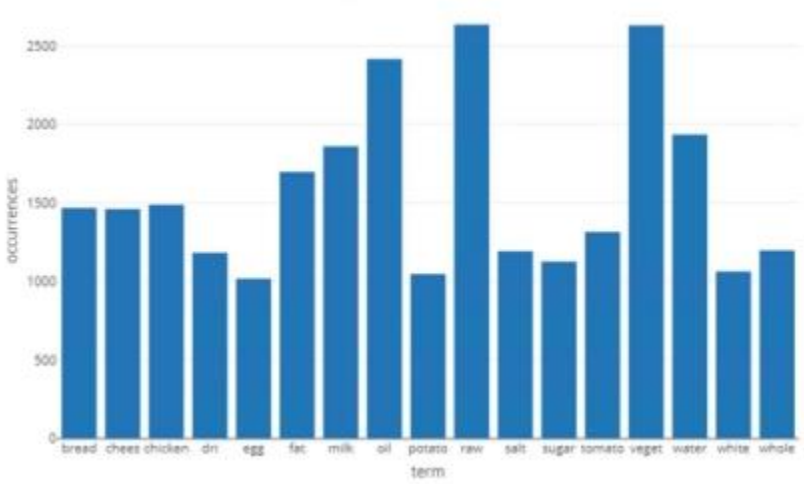


Figure 2: Frequent Text Chunks in Data (1,000 occurrences)

$$False\ Discovery\ Rate\ (FDR) = E\left(\frac{\#\ False\ Positives}{total\ number\ of\ selected\ features}\right) \quad (1)$$

Where E is the expectation and the given ratio is False Discovery Proportion.

Formulae for Gini Index has been mentioned in equation below, which primarily assist in final attributes selection on the basis of variable importance:

$$GI = \sum_k p_k(1-p_k) = 1 - \sum_k p_k^2 \quad (2)$$

where k represents how many classes are there.

4.4. Data Normalization

Data normalization was then used to further process the data set with the finalized features in order to assist achieve improved classification accuracy and prevent the effects of extreme numerical ranges [37, 38]. The following is how the features were scaled:

$$V' = \frac{V-Min}{Max-Min'} \quad (3)$$

where Min and Max values depict the upper and lower data boundaries, and V′ is the scale value that corresponds to the initial value V. Ultimately, the most noteworthy characteristics to be applied to both supervised and unsupervised classifications were those that intersected throughout the two-step variable selection processes.

4.5. Classification

The dimensions of the data were investigated using four different forms of unsupervised machine learning for nonlinear relationships: uniform manifold approximation, factor analysis (FA), t-distributed stochastic Neighbor embedding (t-SNE), Apriori association rules, principal component analysis (PCA), and projection (UMAP) [38]. High-dimensional data may be embedded into lower-dimensional spaces with the help of the t-SNE technique, which is a renowned ML approach for nonlinear dimensionality reduction. For each pair (xi, xj), t-SNE calculates the probabilities pi,j that are proportionate to their corresponding similarities, pj|i, if the high dimensional data (N D) is x_1, x_2, … x_N.

$$p_{j|i} = \frac{exp\left(\frac{-\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} exp\left(\frac{-\|x_i - x_k\|^2}{2\sigma_i^2}\right)} \tag{4}$$

The kNN classifier does computations in two phases. specified a certain similarity metric d, a new testing case x, and a specified k.

- Computes d(x, y) after running through the entire training dataset (y). Let A stand for the k points in the training data y that are closest to x.
- Calculates the proportion of points in A that have a certain class label, or the conditional probability for each class. If an indicator function, is I(z).

$$P(y = j|X = x) = \frac{1}{k}\sum_{i \epsilon A} I(y^{(i)} = j) \tag{5}$$

Based on the model visualization, unsupervised procedures were assessed as the most effective means of assessing the models' appropriateness. In contrast, the ML-classifiers employed certain parameters. Automatic parameter tweaking has been employed with the assistance of repeated technique set at the caret package. Ten rounds of a 15-folded cross-validation resampling were conducted [39]. Based upon the results obtained from k-folds validation, confusion matrix calculation is done, which further assist in gauging metrics like 1) specificity, 2) accuracy, 3) sensitivity and 4) kappa. The performance of the ML-model classifiers was assessed using these metrics. The following formula was used to determine these measures:

$$sensitivity = \frac{TP}{TP+FN} \tag{6}$$

$$specificity = \frac{TN}{TN+FP} \tag{7}$$

$$kappa = \frac{P(a) - P(e)}{1 - P(e)} \tag{8}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

TP in above formulae depicts the number of instances that are correctly classified as positive/Yes class, while ***TN*** depicts the number of instances that are correctly categorized as negative/No class. On the other hand, the instances that are mis-categorized as positive/yes class are referred as ***FP*** and the count instances that are misclassified as negative class are termed as ***FN***.

5. Results and Discussion

Ten noteworthy factors (Fig. 3) that are significant contributors to CRC were identified from the common characteristics obtained from the variable selection techniques. These variables are, in order important factors, including age, total fat, cholesterol, fibre, vitamin E, monounsaturated and saturated fats, carbs, and vitamin B12. The following stage of machine learning modelling made use of these attributes.

The neural network seems to function better than the others. We mapped out the network schematic, and concluded that the employed neural network classification model with a single layer having three hidden nodes was performed best when weight decay was taken into consideration. Additionally, sensitivity analysis identified seven characteristics in the neural network model that should be considered going forward.

In this work, we demonstrate that supervised and unsupervised machine learning techniques may be used to predict colorectal cancer based on a list of significant dietary data. The current study's high prediction accuracy is consistent with other research showing that misclassification rates only varied between 1% and 2% [40, 41]. These machine learning algorithms can be used to forecast the clinical outcomes of colorectal cancer as well as to identify people at risk early on [42, 43]. One of the most effective preventative and changeable strategies for cancer that the public may use is dietary management. Dietary characteristics, such as those of the distal colon and rectum, might provide indicators of the risk of developing certain types of colon-rectal cancer early on [44]. Indeed, a comprehensive analysis of research conducted over 17 years found that there is substantial evidence connecting dietary variables to the incidence of colorectal cancer (CRC). There were few elements in this connection [45].

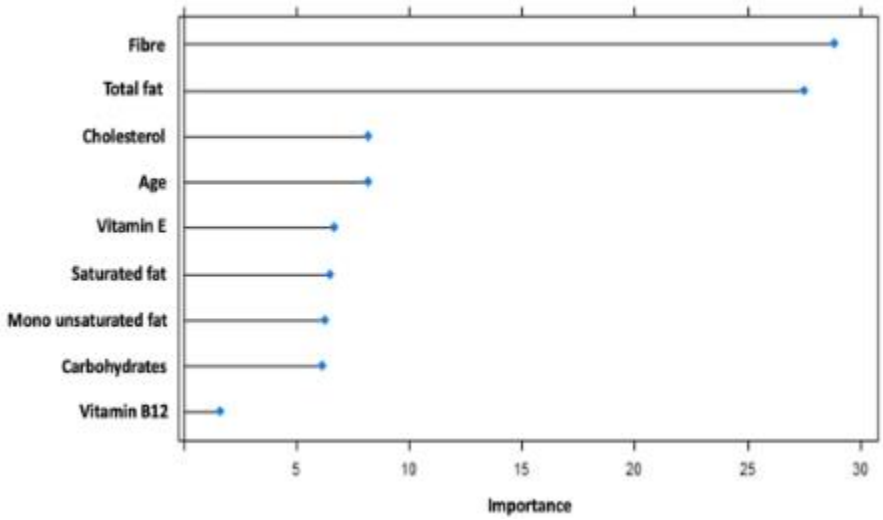


Figure 3: A variable significance plot that illustrates how factors contribute to the prediction of colorectal cancer

Table 2: Two-factor dimensionality reduction model results

Factor based Analysis	Two Factor Model	
	Factor 1	**Factor 2**
Age	0.8	
Carbohydrates	-0.1	0.97
Energy	0.4	0.5
Total fat	0.99	0.1
Fiber	-0.1	0.7
Omega-6	0.5	
Mono unstructured fats	0.9	0.1
Vitamin B12	0.2	0.2
Cholesterol	0.5	
Colorectal cancer	0.65	
Linoleic acid	0.6	

The current investigation found that dietary characteristics that were moderately to highly connected with positive colorectal cancer included total fat, monounsaturated fats, linoleic acid, cholesterol, and omega-6. On the other hand, there is a negative association between colorectal cancer incidences and fiber and carbs. These characteristics are consistent with precision nutrition research showing that dietary parameters, especially those included in the healthy eating index (1) saturated fats, 2) whole fruit, and 3) grains), are more accurate than those found in a single dietary index (like the glycaemic index) when it comes to modifiable behavior for cancer prevention [43, 46]. Furthermore, our apriori algorithm and text mining revealed that 1) vegetables, 2) margarine, 3) eggs, and 4) cheese had significant effects on colorectal cancer [47].

This study's strength is its extensive datasets, which include instances from seven major nations. Owing to processing limitations, we generated estimates, model fts, and classification predictions by randomly sampling observations. Confounding effects might arise because some of the less prevalent elements have to be left out of the model's development.

The CRC outcome label is based on instances that have been discovered and may not reflect risk stratification in different stages and kinds of the disease or early, new, or delayed start of CRC [48]. However, this study has identified key elements that future researchers may take into consideration in a more comprehensive manner, especially multi-dimensional approaches that concurrently take genetics, lifestyle, nutrition, and other relevant aspects into account for the prediction of colorectal cancer [49].

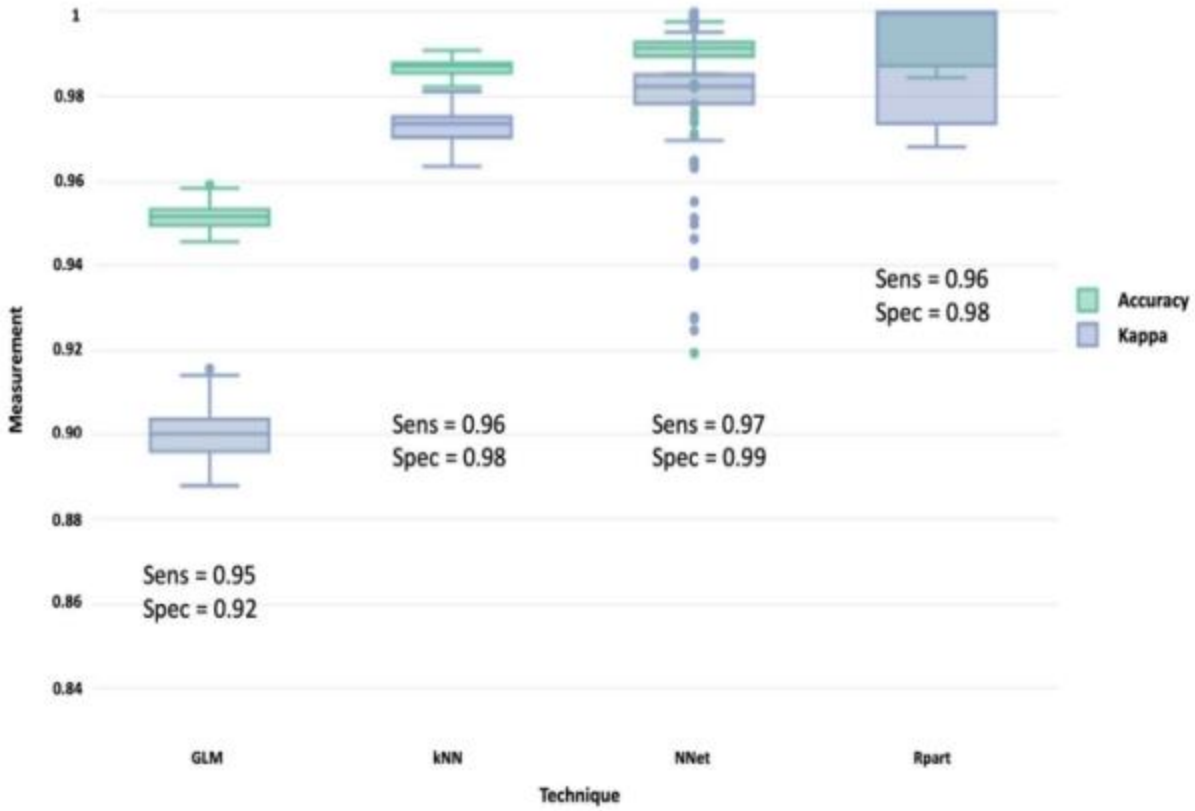


Figure 4: Using dietary data, box plots for the evaluation of performance metrics for colorectal cancer classification models

6. Conclusion

We concluded in this work that the critical dietary factors for colorectal cancer prediction may be explored using a combination of supervised and unsupervised machine learning techniques. With a 1% misclassification of colorectal cancer and a 3% misclassification of non-CRC, the artificial neural network was determined to be the best algorithm for more practical and feasible cancer screening methods. Additionally, using dietary data for screening is a non-invasive method that may be employed on a broad population. Therefore, the success rate of cancer prevention will be significantly increased by using optimum algorithms in conjunction with high cancer screening compliance. Future research should focus on integrating multimodal data, combining gene expression profiles with medical imaging to enhance colorectal cancer prediction accuracy. The use of advanced deep learning architectures, such as transformer-based models, can further improve feature extraction and interpretation in medical diagnostics. Additionally, explainable AI techniques should be explored to increase the transparency and trustworthiness of ML models in healthcare. Developing cost-effective and scalable ML frameworks for early disease detection, particularly in resource-limited settings, will be crucial. Lastly, large-scale, diverse datasets and federated learning approaches can help address data privacy concerns while improving model generalizability across different populations.

Funding Statement: The authors declare that this work was carried out without financial support from any funding agency.

Conflicts of Interest: The authors of this paper have no potential conflicts of interest.

Data Availability: The dietary-related colorectal cancer data are publicly available from the CDC, the Global Dietary Database, and institutional websites

References

[1] Hassibi, K. "Machine learning vs. traditional statistics: Different philosophies, different approaches-DataScienceCentral. com." *Data Science Central* (2016).

[2] Stewart, Matthew. "The Actual Difference Between Statistics and Machine Learning." *Medium: TDS Archive.* https://medium.com/data-science/the-actual-difference-between-statistics-and-machine-learning-64b49f07ea3

[3] Bray, Freddie, Jacques Ferlay, Isabelle Soerjomataram, Rebecca L. Siegel, Lindsey A. Torre, and Ahmedin Jemal. "Global cancer statistics 2018: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries." *CA: a cancer journal for clinicians* 68, no. 6 (2018): 394-424.

[4] Xi, Yue, and Pengfei Xu. "Global colorectal cancer burden in 2020 and projections to 2040." *Translational oncology* 14, no. 10 (2021): 101174.

[5] World Health Organization, Cancer, (2022). Retrieved 20 April 2022 from https://www.who.int/news-room/fact-sheets/detail/cancer.

[6] Bénard, Florence, Alan N. Barkun, Myriam Martel, and Daniel von Renteln. "Systematic review of colorectal cancer screening guidelines for average-risk adults: Summarizing the current global recommendations." *World journal of gastroenterology* 24, no. 1 (2018): 124.

[7] Schreuders, Eline H., Arlinda Ruco, Linda Rabeneck, Robert E. Schoen, Joseph JY Sung, Graeme P. Young, and Ernst J. Kuipers. "Colorectal cancer screening: a global overview of existing programmes." *Gut* 64, no. 10 (2015): 1637-1649.

[8] Araghi, Marzieh, Isabelle Soerjomataram, Aude Bardot, Jacques Ferlay, Citadel J. Cabasag, David S. Morrison, Prithwish De et al. "Changes in colorectal cancer incidence in seven high-income countries: a population-based study." *The lancet Gastroenterology & hepatology* 4, no. 7 (2019): 511-518.

[9] Guren, Marianne Grønlie. "The global challenge of colorectal cancer." *The Lancet Gastroenterology & Hepatology* 4, no. 12 (2019): 894-895.

[10] Dekker, Evelien, Pieter J. Tanis, J. L. Vleugels, Pashtoon M. Kasi, and Michael Wallace. "Pure-amc." *Lancet* 394, no. 10207 (2019): 1467-1480.

[11] Alboaneen, Dabiah, Razan Alqarni, Sheikah Alqahtani, Maha Alrashidi, Rawan Alhuda, Eyman Alyahyan, and Turki Alshammari. "Predicting colorectal cancer using machine and deep learning algorithms: Challenges and opportunities." *Big Data and Cognitive Computing* 7, no. 2 (2023): 74.

[12] Hossain, Md Jakir, Utpala Nanda Chowdhury, M. Babul Islam, Shahadat Uddin, Mohammad Boshir Ahmed, Julian MW Quinn, and Mohammad Ali Moni. "Machine learning and network-based models to identify genetic risk factors to the progression and survival of colorectal cancer." *Computers in Biology and Medicine* 135 (2021): 104539.

[13] Zhao, Dandan, Hong Liu, Yuanjie Zheng, Yanlin He, Dianjie Lu, and Chen Lyu. "A reliable method for colorectal cancer prediction based on feature selection and support vector machine." *Medical & biological engineering & computing* 57, no. 4 (2019): 901-912.

[14] Bingham, Sheila A., Nicholas E. Day, Robert Luben, Pietro Ferrari, Nadia Slimani, Teresa Norat, Françoise Clavel-Chapelon et al. "Dietary fibre in food and protection against colorectal cancer in the European Prospective Investigation into Cancer and Nutrition (EPIC): an observational study." *The lancet* 361, no. 9368 (2003): 1496-1501.

[15] Keum, NaNa, and Edward Giovannucci. "Global burden of colorectal cancer: emerging trends, risk factors and prevention strategies." *Nature reviews Gastroenterology & hepatology* 16, no. 12 (2019): 713-732.

[16] Murphy, Neil, Victor Moreno, David J. Hughes, Ludmila Vodicka, Pavel Vodicka, Elom K. Aglago, Marc J. Gunter, and Mazda Jenab. "Lifestyle and dietary environmental factors in colorectal cancer susceptibility." *Molecular aspects of medicine* 69 (2019): 2-9.

[17] Shafi, A. S. M., MM Imran Molla, Julakha Jahan Jui, and Mohammad Motiur Rahman. "Detection of colon cancer based on microarray dataset using machine learning as a feature selection and classification techniques." *SN Applied Sciences* 2, no. 7 (2020): 1243.

[18] Islam, Ashraful, Mohammad Masudur Rahman, Eshtiak Ahmed, Faisal Arafat, and Md Fazle Rabby. "Adaptive feature selection and classification of colon cancer from gene expression data: an ensemble learning approach." In *Proceedings of the international conference on computing advancements*, pp. 1-7. 2020.

[19] Bae, Jin Hee, Minwoo Kim, J. S. Lim, and Zong Woo Geem. "Feature selection for colon cancer detection using k-means clustering and modified harmony search algorithm." *Mathematics* 9, no. 5 (2021): 570.

[20] Hamida, A. Ben, Maxime Devanne, Jonathan Weber, Caroline Truntzer, Valentin Derangère, François Ghiringhelli, Germain Forestier, and Cédric Wemmert. "Deep learning for colon cancer histopathological images analysis." *Computers in Biology and Medicine* 136 (2021): 104730.
[21] Al-Rajab, Murad, Joan Lu, and Qiang Xu. "A framework model using multifilter feature selection to enhance colon cancer classification." *Plos one* 16, no. 4 (2021): e0249094.
[22] Wang, Pu, Xiao Xiao, Jeremy R. Glissen Brown, Tyler M. Berzin, Mengtian Tu, Fei Xiong, Xiao Hu et al. "Development and validation of a deep-learning algorithm for the detection of polyps during colonoscopy." *Nature biomedical engineering* 2, no. 10 (2018): 741-748.
[23] Rajesh, Gundlapalle, Boda Saroja, M. Dhivya, and A. B. Gurulakshmi. "DB-scan algorithm based colon cancer detection and stratification analysis." In *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*, pp. 644-648. IEEE, 2020.
[24] Jørgensen, Alex Skovsbo, Anders Munk Rasmussen, Niels Kristian Mäkinen Andersen, Simon Kragh Andersen, Jonas Emborg, Rasmus Røge, and Lasse Riis Østergaard. "Using cell nuclei features to detect colon cancer tissue in hematoxylin and eosin stained slides." *Cytometry Part A* 91, no. 8 (2017): 785-793.
[25] Rahman, Md Akizur, and Ravie Chandren Muniyandi. "Feature selection from colon cancer dataset for cancer classification using artificial neural network." *Int. J. Adv. Sci. Eng. Inf. Technol* 8, no. 4-2 (2018): 1387-1393.
[26] Choi, Seong Ji, Eun Sun Kim, and Kihwan Choi. "Prediction of the histology of colorectal neoplasm in white light colonoscopic images using deep learning algorithms." *Scientific Reports* 11, no. 1 (2021): 5311.
[27] Yao, Yao, Shuiping Gou, Ru Tian, Xiangrong Zhang, and Shuixiang He. "Automated Classification and Segmentation in Colorectal Images Based on Self-Paced Transfer Network." *BioMed Research International* 2021, no. 1 (2021): 6683931.
[28] Centers for Disease Control and Prevention, National Health and Nutrition Examination Survey, (2022). Retrieved 20 April 2022 from https://www. cdc.gov/nchs/nhanes/index.htm.
[29] Global Dietary Database, Microdata Surveys, (2018). Retrieved March 2022 from https://www.globaldietarydatabase.org/management/micro data-surveys.
[30] U.S. National Library of Medicine, National Center for Biotechnology Information: dbGAP data, (2022). Retrieved March 2022 from https://www. ncbi.nlm.nih.gov/projects/gap/cgi-bin/collection.cgi?study_id=phs001991.v1.p1.
[31] Inter-university Consortium for Political and Social Research, Find Data, (2022). Retrieved March 2022 from https://www.icpsr.umich.edu/web/ pages/.
[32] China Health and Nutrition Survey, China Health and Nutrition Survey, (2015). Retrieved March 2022 from https://www.cpc.unc.edu/projects/ china.
[33] Government of Canada, Canadian Community Health Survey, (2018). Retrieved March 2022 from https://www.canada.ca/en/health-canada/ services/food-nutrition/food-nutrition-surveillance/health-nutritionsurveys/canadian-community-health-survey-cchs.html.
[34] Data.world, Data.world, (2022). Retrieved March 2022 from https://ourwo rldindata.org.
[35] Ripley, Brian, Bill Venables, Douglas M. Bates, Kurt Hornik, Albrecht Gebhardt, David Firth, and Maintainer Brian Ripley. "Package 'mass'." *Cran r* 538, no. 113-120 (2013): 822.
[36] Kursa, Miron B., and Witold R. Rudnicki. "Feature selection with the Boruta package." *Journal of statistical software* 36 (2010): 1-13.
[37] Zhao, Mingyuan, Chong Fu, Luping Ji, Ke Tang, and Mingtian Zhou. "Feature selection and parameter optimization for support vector machines: A new approach based on genetic algorithm with feature chromosomes." *Expert Systems with Applications* 38, no. 5 (2011): 5197-5204.
[38] Dinov, Ivo D. "Data science and predictive analytics." *Cham, Switzerland: Springer* (2018).
[39] Kuhn, Max, Jed Wing, Steve Weston, Andre Williams, Chris Keefer, Allan Engelhardt, Tony Cooper, Zachary Mayer, Brenton Kenkel, and R. Core Team. "Package 'caret'." *The R Journal* 223, no. 7 (2020): 48.
[40] Nartowt, Bradley J., Gregory R. Hart, Wazir Muhammad, Ying Liang, Gigi F. Stark, and Jun Deng. "Robust machine learning for colorectal cancer risk prediction and stratification." *Frontiers in big Data* 3 (2020): 6.
[41] Hornbrook, Mark C., Ran Goshen, Eran Choman, Maureen O'Keeffe-Rosetti, Yaron Kinar, Elizabeth G. Liles, and Kristal C. Rust. "Early colorectal cancer detected by machine learning model using gender, age, and complete blood count data." *Digestive diseases and sciences* 62, no. 10 (2017): 2719-2727.
[42] Gründner, Julian, Hans-Ulrich Prokosch, Michael Stürzl, Roland Croner, Jan Christoph, and Dennis Toddenroth. "Predicting clinical outcomes in colorectal cancer using machine learning." In *Building Continents of Knowledge in Oceans of Data: The Future of Co-Created eHealth*, pp. 101-105. IOS Press, 2018.

[43] Shiao, S. Pamela K., James Grayson, Amanda Lie, and Chong Ho Yu. "Personalized nutrition—genes, diet, and related interactive parameters as predictors of cancer in multiethnic colorectal cancer families." *Nutrients* 10, no. 6 (2018): 795.
[44] Hofseth, Lorne J., James R. Hebert, Anindya Chanda, Hexin Chen, Bryan L. Love, Maria M. Pena, E. Angela Murphy et al. "Early-onset colorectal cancer: initial clues and current views." *Nature reviews Gastroenterology & hepatology* 17, no. 6 (2020): 352-364.
[45] Tabung, Fred K., Lisa S. Brown, and Teresa T. Fung. "Dietary patterns and colorectal cancer risk: a review of 17 years of evidence (2000–2016)." *Current colorectal cancer reports* 13, no. 6 (2017): 440-454.
[46] Li, Tian, Chunqiu Zheng, Le Zhang, Ziyuan Zhou, and Rong Li. "Exploring the risk dietary factors for the colorectal cancer." In *2015 IEEE International Conference on Progress in Informatics and Computing (PIC)*, pp. 570-573. IEEE, 2015.
[47] Zuhri, Mohammad AZ Abu, Mohammed Awad, Shahnaz Najjar, Nuha El Sharif, and Issa Ghrouz. "Colorectal cancer risk factor assessment in Palestine using machine learning models." *International Journal of Medical Engineering and Informatics* 16, no. 2 (2024): 126-138.
[48] Zheng, Ling, Elijah Eniola, and Jiacun Wang. "Machine learning for colorectal cancer risk prediction." In *2021 International Conference on Cyber-Physical Social Intelligence (ICCSI)*, pp. 1-6. IEEE, 2021.
[49] Jørgensen, Alex Skovsbo, Anders Munk Rasmussen, Niels Kristian Mäkinen Andersen, Simon Kragh Andersen, Jonas Emborg, Rasmus Røge, and Lasse Riis Østergaard. "Using cell nuclei features to detect colon cancer tissue in hematoxylin and eosin stained slides." *Cytometry Part A* 91, no. 8 (2017): 785-793.

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Traffic Sign Recognition Using a Customized Convolutional Neural Network

Haseeb Gul[1, *], Moiz Gul[2]

[1] Department of Computer Science, Bahauddin Zakariya University Multan, 60000, Pakistan
[2] Department of Information Technology, Emerson University Multan, 60000, Pakistan
[*]Corresponding Author: Haseeb Gul. Email: haseebcs442@gmail.com

Received: 20 November 2024; Revised: 27 December 2025; Accepted: 7 February 2025; Published: 20 March 2025
AID: 004-01-000049

Abstract: Recognition of traffic signs is fundamentally a multiclass classification problem that is an essential part of autonomous driving system that aid vehicles recognize and obey road regulations. Due to their power to learn and generalize from data, neural networks have become a useful approach for solving complex problems presented by image classification tasks. Such systems are considered as a major component of an intelligent transportation system which enhances road safety and prevents from potential hazards. In the underlying research study, a customized Convolutional Neural Network (CNN) has been exploited for the categorization of traffic signs based on the German Traffic Sign Recognition Benchmark (GTSRB) dataset. The proposed model integrated with data augmentation and a robust architecture of CNN excels with an overwhelming accuracy of approximately 97% and can easily be deployed in real products like intelligent and automated traffic management systems, road safety solutions and self-driven vehicles etc.

Keywords: Traffic Signs Classification; Advance Driving Assistance Systems; Convolutional Neural Network; Deep Learning;

1. Introduction

Deep learning based advanced models, especially neural networks have significantly transformed the landscape of artificial intelligence and machine learning by providing highly efficient as well as optimal solutions to different intricated problems. Convolutional Neural Networks (CNNs) pioneered by LeCun et al. in their seminal work on gradient-based learning [1] represents a specialized architecture that excels at hierarchical feature extraction from visual data. The structure and functioning of human brain serve as a foundation for the development of neural networks [2] which excel at learning representations of sequential data by allowing them to carry out tasks such as speech recognition, image processing, image detection and natural language processing. The specialized network architecture of CNNs' represents an advanced method to image classification because these networks successfully extract hierarchical as well as spatial features from images.

The autonomous driving system utilizes traffic sign recognition as a means for CNNs to have meaningful applications across different fields. Benchmark studies by Stallkamp et al. [3] established the German Traffic Sign Recognition Benchmark (GTSRB) as the standard evaluation framework, while Ciresan et al., [4] demonstrated unprecedented accuracy using multi-column deep neural networks. Autonomous vehicles use traffic sign recognition as a core requirement in intelligent transportation systems because of its complexity. By identifying and classifying traffic signs accurately these systems incorporate in enhancing

road safety, improving the traffic flow and aid to support the development of autonomous or self-driving technologies. The integration of traffic sign recognition into real-world applications underwits the potential of neural networks in addressing challenges faced in the society due to traffic and vehicle driving.

By incorporating the German Traffic Sign Recognition Benchmark (GTSRB) dataset, the paper focuses on the recognition of traffic signs. The primary focus is to develop and evaluate a model based on CNN capable of acquiring high classification accuracy to all across 43 distinct classes of traffic signs mentioned in the dataset. The distinctive contributions of this study include the implementation of an augmented training strategy and a robust CNN architecture has been designed to enhance the generalization as well as handling the imbalances in the class efficiently. This study also highlights a wider analysis of the model's performance also demonstrating its practical applications.

2. Literature Review

Traffic Sign Recognition (TSR) is a critical element of an Advanced Driver Assistance Systems (ADAS) and an intelligent traffic management system that focuses on improving road safety and the flow of traffic on the road. It consists of two primary stages; one is the traffic sign detection i.e localization of signs within an image and the other one is the recognition of traffic sign which is the classification of the detected signs. The accuracy obtained by the detection directly impacts the final recognition. Previously used methods for TSR often relied on handcrafted features like shape, color and texture. However, these techniques and methods can affect easily by environmental factors such as weather conditions, illumination changes, occlusions as well as sign aging [5], [6]. Advanced Neural Networks, particularly CNNs are regarded as an authentic tool to tackle with the challenges faced by automatically learning hierarchical and sequential invariant features from raw pixel data [4].

Different studies through light on the application of CNNs to traffic sign recognition, demonstrating their ability to achieve high accuracy less calculation of the loss. Existing works involved the use of basic CNN architectures like LeNet-5, often modified or improved for the task of TSR. For example, a few numbers of researchers have performed experiments using Gabor kernels as initial convolutional kernels in LeNet-5 [7]. Some researchers have added layers batch normalization after pooling layers or have used different types of optimizers like Adam [8]. However, standard CNN architectures like LeNet-5 and ResNet exhibit critical limitations in real-world deployment. Despite high accuracy in controlled settings, these models lack robustness to environmental conditions like motion blur, lighting variations and weather induced distortions [9],[10] severely restricting their reliability in autonomous driving systems. This shortfall arises because real-world traffic sign images frequently deviate from idealized dataset conditions due to dynamic occlusions patterns, non-uniform illumination changes and perspective distortions during vehicle motion [6],[11].

To explore and investigates the limitations of basic architectures of CNN, recent studies have demonstrated and explored the use of a deeper and more complex CNN models, such as that of a Multi-Column Deep Neural Networks (MCDNNs), which combine various CNNs trained on differently preprocessed data [4]. Additionally, other studies explored and demonstrates how transfer learning can be beneficial in training a CNN for TSR using a small number of standard traffic training examples [12]. Some researchers have often experimented with other CNN architectures e.g. ResNet and Capsule Networks or even combinations of CNNs with different other machine learning models like SVMs or ELMs and with different types of algorithms such as AdaBoost [13, 14]. The YOLO family architecture family which is also known as (You Only Look Once) includes YOLOv3 and YOLOv4, have commonly been adopted in the field of TSR because of their high efficiency and accuracy. e.g., using YOLOv3 [15] and YOLOv4 [16] for real time TSR. Other work is based on previous studies in which synthetic training data generated by GANs is incorporated to improve YOLO for TSR [17]. A few researchers implementing a lightweight version of YOLO like YOLOv4-tiny for embedded and ensemble systems [16]. Moreover, practical versions of YOLO used with alternate tracking algorithms can also be found [18].

A different perspective of the research involves a customized CNN architecture that contains specific modifications and amendments to TSR. For instance, researchers have explored the use of Mask R-CNN

detector both for the recognition and detection and have made suitable improvements that enhances the rate of recall, particularly for the detection of small traffic signs [19]. In addition to some researchers have used a customized CNN with multiple outputs in the final neural network layer. It consists of both a regression output for traffic sign coordinates and a classification output for classifying the type of sign [20]. These types of model architectures that are customized have been found with a high performance and high accuracy on complex and intricated datasets with similar categories and low variability among distinct categories [21].

A crucial and necessary element of TSR based on deep learning is obtaining huge and diverse datasets which can be easily represented. Various researchers incorporate the use of German Traffic Sign Recognition Benchmark (GTSRB) as their foundation because this benchmark serves as a standard benchmark for the analysis and evaluation of TSR algorithms [3], [22]. Researchers modify their self-collected real-world datasets using the techniques like geometric mean and the appearance of distortion to enhance the dataset diversity and validation of its effectiveness [8]. Whereas, most of the studies have used a dataset with static images. The evaluation of the TSR models require video testing using scenarios in the real-world on embedded systems which serves for the practical evaluations [23], [24].

In conclusion, the literature on TSR using customized CNN architectures reflects ongoing advancements toward robust and real-time monitoring systems. Progress has been driven by innovations in network architecture, training strategies, and data augmentation methods. The research activities have significant success in attaining impressive accuracy and speed among different datasets. However, further efforts are required to be made to address the challenges associated with real-time application systems. [11], [25].

3. Methodology

This section provides a methodological overview adopted for the development and evaluation of CNN model trained on customized model architecture for the recognition of traffic signs. It sheds light on the dataset used, preprocessing techniques applied, the proposed model architecture, training and the evaluation metrices used. By acquiring a systematic approach, this methodology ensures that the model is both accurate and generalized in the recognition of different varieties of traffic signs. A thorough overview of the methodology is demonstrated in the following section.

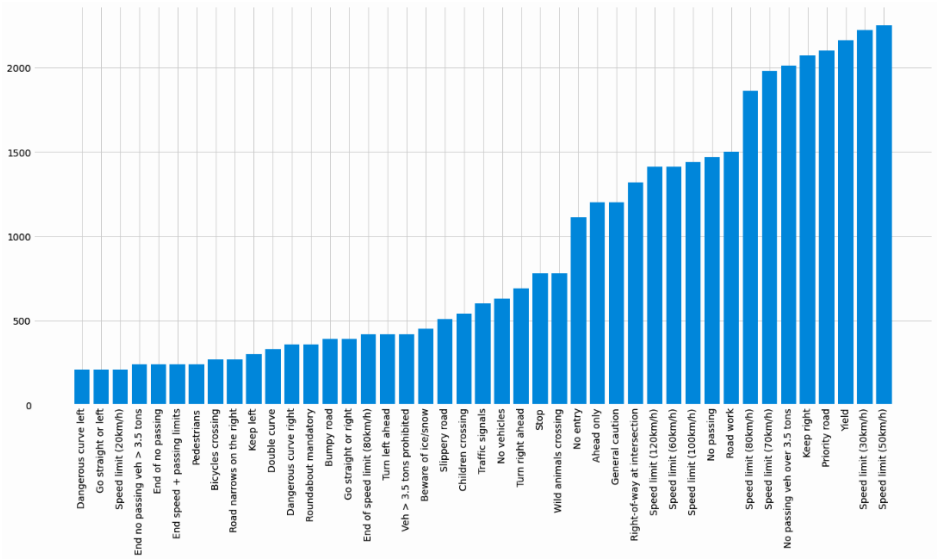
3.1. Dataset and Preprocessing Methods

The dataset used for the study is the German Traffic Sign Recognition Benchmark (GTSRB) [3]. There are 43 distinct classes of traffic signs in the dataset which are comprising a total of 51,839 images. The dataset is splitted into 36,287 training images and 12,630 testing images. The original image dimensions vary between 15 × 15 pixels and 250 × 250 pixels. To ensure the computational efficiency hence preserving important visual features, all images were resized to 30 × 30 pixels with RGB channels, providing compatibility with the model input layer [22]. To overcome the challenges of class imbalance, enhance model generalization and increase the diversity of dataset, different techniques for data augmentation were incorporated. These included zoom (0.2), rotation (±15°), height and width shifts (0.1), shear transformation (0.2) and horizontal and vertical translations. This whole process generates five augmented samples per original image which aid in expanding the effective dataset size to 181,435 samples. Such augmentation mitigates the impact of real-world environmental variations such as illumination shifts, viewpoint changes and partial occlusions that are critical challenges for robust TSR in autonomous driving scenarios [6, 8].

3.2 Dataset Analysis

The dataset encompasses an enormous number of images which are organized into 43 classes in the category of traffic signs. Complex and extensive data augmentation substantially increased the effective dataset size, thereby improving the capacity of the model to generalize with new unseen data. Figure 1 shows the proportion of images distributed across different classes, illustrating the initial imbalance and the manner in which augmentation addressed this disparity.

Figure 1: Dataset Analysis



3.3. Architecture of Model

The proposed model is a sequential CNN model as shown in figure 2 consisting of the layers given below:

- A max-pooling layer (2 x 2) is placed after two convolutional layers (32 filters, 3 x 3 kernel) using the ReLU activation function.
- A max-pooling layer (2 x 2) with further two convolutional layers (64 filters, 3x3 kernel) with ReLU activation function.
- A dense layer consisting of 512 neurons with ReLU activation is placed after the flattening layer.
- To prevent overfitting dropout layer (0.5 rate) is incorporated.
- The output layer is a dense layer with 43 neurons and Softmax activation function for a multi-class classification.

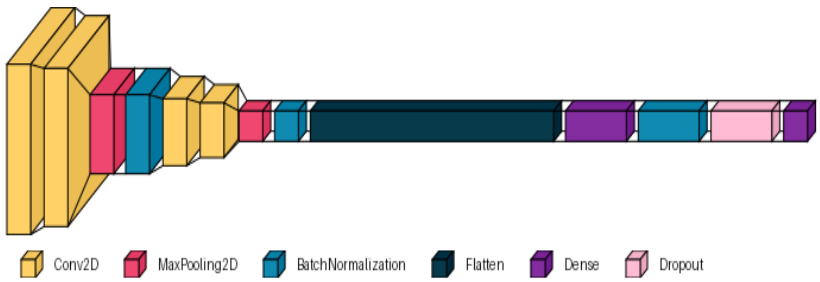


Figure 2: Model Architecture

Adam optimizer is employed in the compilation of the model together with categorical cross-entropy loss with a learning rate of 0.001. Training was conducted on approximately 20 epochs with a batch size of 32 on NVIDIA RTX 3090 GPU (24 GB VRAM), utilizing a total training time of 120 minutes.

The design of this architecture follows the VGG-style convolutional stacking approach [26], but with reduced depth to accommodate low-resolution inputs (30 × 30 pixels), thereby avoiding excessive down-sampling that could degrade small sign features. Alternative architectures, such as ResNet-18 and LeNet-5

[7], were also evaluated but were discarded due to longer computational times and marginal accuracy improvements (<0.5%). The relatively shallow depth of the proposed model ensures computational efficiency for low-resolution inputs while maintaining strong discriminative capability.

3.4. Evaluation Metrics

Standard metrics of classification were used for the performance evaluation of the advocated model, which encompasses the metrices such as Precision, Recall, F1-score and Support, as illustrated in Table 1. A confusion matrix was developed to investigate the possible errors of the model enabling a thorough analysis of the misclassification in the patterns which are shown in Figure 4.

Moreover, the training progress of the undertaken model was evaluated by plotting accuracy, validation accuracy and cross-entropy loss graph over epochs. This graphical representation shows a clear view into the learning dynamics and convergence behavior of the model. The classification reports containing the combined scores of Precision, Recall, and F1-score of all classes is shown in table 1.

Table 1: Classification Report

Class_id	Precision	Recall	F1-score	Support
0	0.79	1.00	0.88	60
1	1.00	1.00	1.00	720
2	0.99	1.00	0.99	750
3	0.96	0.98	0.97	450
4	1.00	1.00	1.00	660
5	0.98	0.99	0.98	630
6	1.00	0.97	0.99	150
7	1.00	1.00	1.00	450
8	1.00	0.97	0.98	450
9	1.00	1.00	1.00	480
10	1.00	1.00	1.00	660
11	0.96	1.00	0.98	420
12	1.00	0.98	0.99	690
13	1.00	1.00	1.00	720
14	1.00	1.00	1.00	270
15	0.99	1.00	0.99	210
16	1.00	1.00	1.00	150
17	1.00	1.00	1.00	360
18	0.99	0.93	0.96	390
19	0.97	1.00	0.98	60
20	0.98	1.00	0.99	90
21	0.83	1.00	0.91	90
22	0.99	0.84	0.91	120
23	0.99	0.99	0.99	150
24	0.97	0.98	0.97	90

25	1.00	0.97	0.98	480
26	0.83	1.00	0.91	180
27	0.86	0.50	0.63	60
28	0.99	0.99	0.99	150
29	0.85	1.00	0.92	90
30	0.99	0.83	0.90	150
31	1.00	1.00	1.00	270
32	1.00	1.00	1.00	60
33	0.99	1.00	1.00	210
34	1.00	1.00	1.00	120
35	0.99	0.98	0.99	390
36	0.98	1.00	0.99	120
37	1.00	0.98	0.99	60
38	1.00	1.00	1.00	690
39	0.99	0.98	0.98	90
40	0.96	0.98	0.97	90
41	1.00	1.00	1.00	60
42	0.99	1.00	0.99	90
Accuracy			**0.98**	**12630**
Macro Avg	0.97	0.97	0.97	12630
Weighted avg	0.99	0.98	0.98	12630

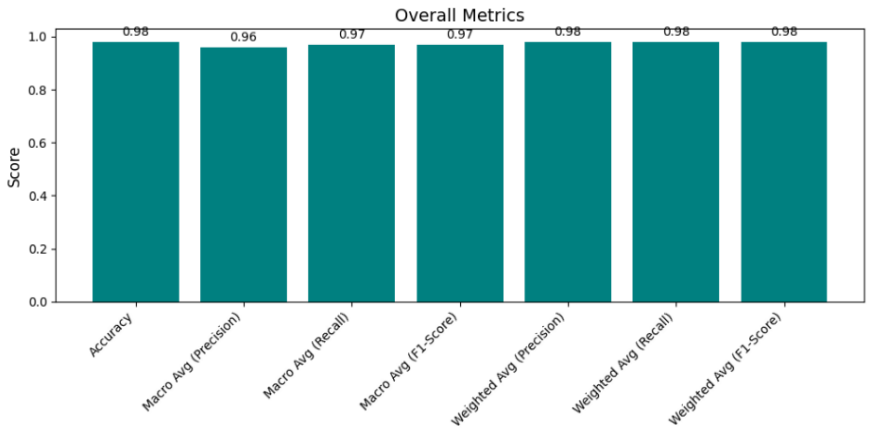


Figure 3: Aggregate performance on metrices (Precision, Recall, F1-score)

4. Findings and Analysis

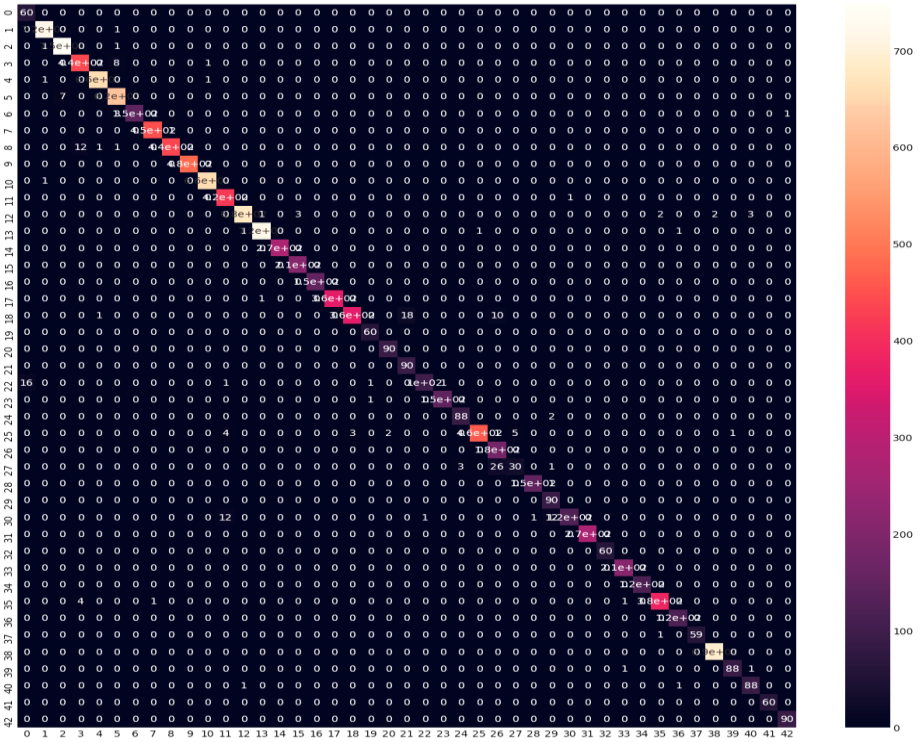
In this section, they exhibit the analysis and application of the outcomes of the operations of the CNN model to be outlined in the research they propose to understand different traffic signs. The current section will demonstrate a description of the dataset and its details. It also gives the assessment of the model efficiency with various statistical metrices: precision, recall and F1-score. The accuracy and loss of the

model, confusion matrix and comparative analysis to explain effectiveness of model are also displayed in this chapter.

4.1 Model Performance

The proposed model has achieved a maximum accuracy of about 97%. Evaluation metrics are summarized displaying the high recall and precision score across in many of the classes. A confusion matrix analysis as shown in figure 4 reveals that the misclassifications primarily occurred among visually similar signs, like as signs of speed limit with minor variation in numerical values. The confusion matrix as developed after the model training and evaluation and is showed below:

Figure 4: Confusion Matrix for 43 traffic sign classes



4.2. Comparative Analysis

When compared with the existing approaches and techniques, the proposed model exhibits high generalization capabilities and accuracy gains. It outperforms baseline models such as ResNet-34 and YOLOv4-tiny in both accuracy and F1-score. These improvements are largely attributed to the integration of different techniques of data augmentation and the design of a compact yet robust CNN architecture optimized for the task.

The quantitative comparison between the model proposed and existing approaches is illustrated in Table 2 which are provided below:

Table 2: Quantitative Comparative Analysis

Model	Accuracy (%)	F1-Score	Reference
Proposed CNN	97.0	0.96	-
ResNet-34	95.2	0.94	[21]
YOLOv4-tiny	93.5	0.92	[16]

4.3. Visualization

The accuracy and loss curves, as shown in figure 5, demonstrate a consistent improvement across the training epochs, with minimum chances of overfitting. The gradual convergence of both training and validation metrics shows that the model has learned effectively without significant performance degradation on unseen data.

In addition to the performance curves, visualizations of predictions on test images further confirms the capability of the model to correctly distinguishes a huge range of traffic sign categories, including those with challenging visual variations such as illumination changes, occlusions, and scale differences.

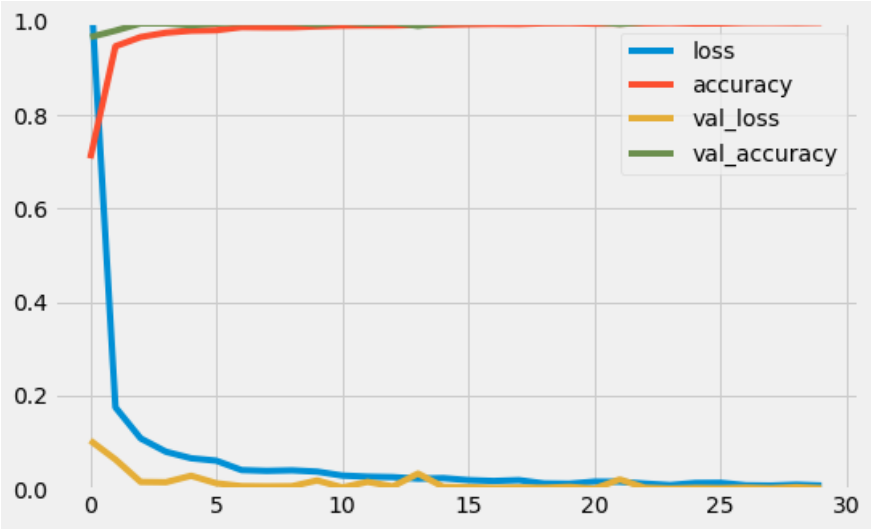


Figure 5: Training and Validation Accuracy and Loss Curves

The variation of validation accuracy and loss over the epochs as presented in Figure 6, providing further evidence of the stable training process and capability of the model for the generalization of the new data.

```
Epoch 1/20
/usr/local/lib/python3.11/dist-packages/keras/src/trainers/data_adapters/py_dataset_adapter.py:122: UserWarning: Your `PyD
  self._warn_if_super_not_called()
858/858 ━━━━━━━━━━━━━━━━━━━━ 133s 148ms/step - accuracy: 0.5208 - loss: 1.9846 - val_accuracy: 0.9692 - val_loss: 0.1012
Epoch 2/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 134s 140ms/step - accuracy: 0.9364 - loss: 0.2126 - val_accuracy: 0.9858 - val_loss: 0.0476
Epoch 3/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 130s 151ms/step - accuracy: 0.9668 - loss: 0.1111 - val_accuracy: 0.9907 - val_loss: 0.0319
Epoch 4/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 130s 137ms/step - accuracy: 0.9659 - loss: 0.1126 - val_accuracy: 0.9888 - val_loss: 0.0339
Epoch 5/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 118s 137ms/step - accuracy: 0.9742 - loss: 0.0843 - val_accuracy: 0.9971 - val_loss: 0.0107
Epoch 6/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 122s 142ms/step - accuracy: 0.9762 - loss: 0.0765 - val_accuracy: 0.9955 - val_loss: 0.0129
Epoch 7/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 140s 139ms/step - accuracy: 0.9813 - loss: 0.0597 - val_accuracy: 0.9874 - val_loss: 0.0474
Epoch 8/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 118s 137ms/step - accuracy: 0.9818 - loss: 0.0568 - val_accuracy: 0.9968 - val_loss: 0.0113
Epoch 9/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 142s 138ms/step - accuracy: 0.9856 - loss: 0.0452 - val_accuracy: 0.9942 - val_loss: 0.0204
Epoch 10/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 121s 141ms/step - accuracy: 0.9859 - loss: 0.0517 - val_accuracy: 0.9921 - val_loss: 0.0286
Epoch 11/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 123s 143ms/step - accuracy: 0.9854 - loss: 0.0445 - val_accuracy: 0.9969 - val_loss: 0.0085
Epoch 12/20
858/858 ━━━━━━━━━━━━━━━━━━━━ 125s 145ms/step - accuracy: 0.9870 - loss: 0.0463 - val_accuracy: 0.9978 - val_loss: 0.0072
Epoch 13/20
```

Figure 6: Validation Accuracy and Loss by Epochs

4.4. Prediction on Test Data

Sample test predictions as demonstrated in Figure 7, presents a comparison between the actual and predicted labels for the selected test images. These visualizations confirm the ability of the model to correctly distinguishes different traffic signs across multiple categories.

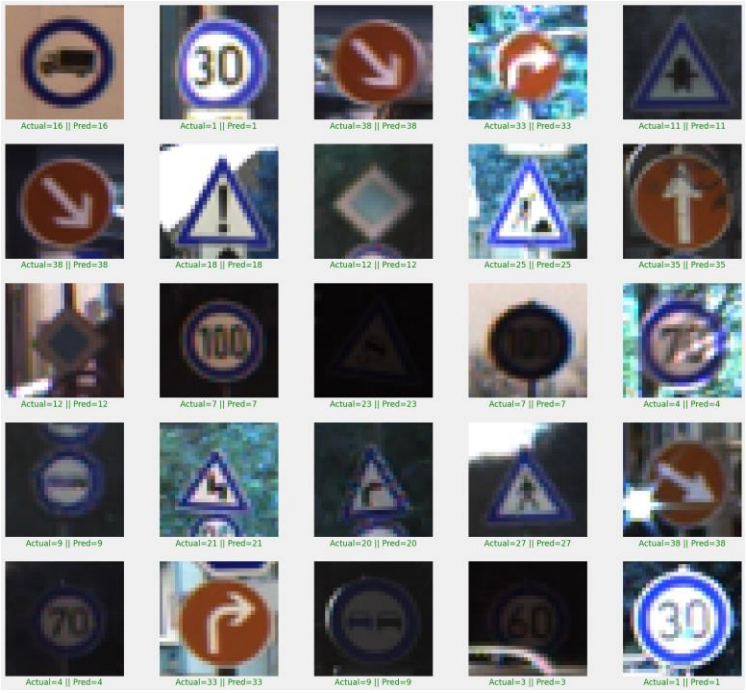


Figure 7: Sample Test Predictions

5. Conclusion

This study demonstrates how effectively Convolutional Neural Networks (CNNs) can act as a robust approach for Traffic Sign Recognition (TSR) using the German Traffic Sign Recognition Benchmark (GTSRB) dataset. The proposed model has achieved a maximum accuracy of approximately 97%, showing its practical potential for deployment in Intelligent Transportation System applications. The research addressed different challenges such as misclassification of visually similar signs by incorporating and class imbalance by incorporating a carefully designed CNN architecture in conjunction with extensive data augmentation techniques. These measures significantly boost the generalization capabilities of model and robustness to real-world conditions. The future research will concentrate on integrating attention mechanisms to further enhance the model's discriminative power and to address persistent misclassification issues in visually similar traffic sign categories. This enhancement is expected to further strengthen the model's applicability in advanced autonomous driving and smart transportation systems.

Funding Statement: No external funding was obtained for the completion of this research.

Conflicts of Interest: No conflicts of interest are associated with this study.

Data Availability: The data used in this study are publicly available from the German Traffic Sign Recognition Benchmark (GTSRB) dataset.

References

[1] LeCun, Yann, Léon Bottou, Yoshua Bengio, and Patrick Haffner. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86, no. 11 (2002): 2278-2324.

[2] Abdi, Lotfi, and Aref Meddeb. "Deep learning traffic sign detection, recognition and augmentation." In *Proceedings of the Symposium on Applied Computing*, pp. 131-136. 2017.

[3] Stallkamp, Johannes, Marc Schlipsing, Jan Salmen, and Christian Igel. "The German traffic sign recognition benchmark: a multi-class classification competition." In *The 2011 international joint conference on neural networks*, pp. 1453-1460. IEEE, 2011.

[4] Dan, Cireşan, Meier Ueli, Masci Jonathan, and Schmidhuber Jürgen. "Multi-column deep neural network for traffic sign classification." *Neural networks* 32, no. 1 (2012): 333-338.

[5] Gudigar, Anjan, Shreesha Chokkadi, and Raghavendra U. "A review on automatic detection and recognition of traffic sign." *Multimedia Tools and Applications* 75, no. 1 (2016): 333-364.

[6] Mogelmose, Andreas, Mohan Manubhai Trivedi, and Thomas B. Moeslund. "Vision-based traffic sign detection and analysis for intelligent driver assistance systems: Perspectives and survey." *IEEE transactions on intelligent transportation systems* 13, no. 4 (2012): 1484-1497.

[7] Ellahyani, Ayoub, Mohamed El Ansari, Redouan Lahmyed, and Alain Trémeau. "Traffic sign recognition method for intelligent vehicles." *Journal of the Optical Society of America A* 35, no. 11 (2018): 1907-1914.

[8] Bangquan, Xie, and Weng Xiao Xiong. "Real-time embedded traffic sign recognition using efficient convolutional neural network." *IEEE Access* 7 (2019): 53330-53346.

[9] Li, Chen, and Cheng Yang. Zeng, Yujun, Xin Xu, Yuqiang Fang, and Kun Zhao. "Traffic sign recognition using deep convolutional networks and extreme learning machine." In *International Conference on Intelligent Science and Big Data Engineering*, pp. 272-280. Cham: Springer International Publishing, 2015.

[10] Zhu, Yanzhao, and Wei Qi Yan. "Traffic sign recognition based on deep learning." *Multimedia Tools and Applications* 81, no. 13 (2022): 17779-17791.

[11] In *2016 16th International Symposium on Communications and Information Technologies (ISCIT)*, pp. 156-161. IEEE, 2016.

[12] Zhu, Yanzhao, and Wei Qi Yan. "Traffic sign recognition based on deep learning." *Multimedia Tools and Applications* 81, no. 13 (2022): 17779-17791.

[13] Qin, Zhongbing, and Wei Qi Yan. "Traffic-sign recognition using deep learning." In *International symposium on geometry and vision*, pp. 13-25. Cham: Springer International Publishing, 2021.

[14] Huang, Zhiyong, Yuanlong Yu, Jason Gu, and Huaping Liu. "An efficient method for traffic sign recognition based on extreme learning machine." *IEEE transactions on cybernetics* 47, no. 4 (2016): 920-933.

[15] Rajendran, Shehan P., Linu Shine, R. Pradeep, and Sajith Vijayaraghavan. "Real-time traffic sign recognition using YOLOv3 based detector." In *2019 10th international conference on computing, communication and networking technologies (ICCCNT)*, pp. 1-7. IEEE, 2019.

[16] Wang, Lanmei, Kun Zhou, Anliang Chu, Guibao Wang, and Lizhe Wang. "An improved light-weight traffic sign recognition algorithm based on YOLOv4-tiny." *Ieee Access* 9 (2021): 124963-124971.

[17] Radu, Mihai Daniel, Ilona Madalina Costea, and Valentin Alexandru Stan. "Automatic traffic sign recognition artificial inteligence-deep learning algorithm." In *2020 12th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp. 1-4. IEEE, 2020.

[18] Yuan, Yuan, Zhitong Xiong, and Qi Wang. "An incremental framework for video-based traffic sign detection, tracking, and recognition." *IEEE Transactions on Intelligent Transportation Systems* 18, no. 7 (2016): 1918-1929.

[19] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems* 28 (2015).

[20] Vennelakanti, Aashrith, Smriti Shreya, Resmi Rajendran, Debasis Sarkar, Deepak Muddegowda, and Phanish Hanagal. "Traffic sign detection and recognition using a CNN ensemble." In *2019 IEEE international conference on consumer electronics (ICCE)*, pp. 1-4. IEEE, 2019.

[21] Tabernik, Domen, and Danijel Skočaj. "Deep learning for large-scale traffic-sign detection and recognition." *IEEE transactions on intelligent transportation systems* 21, no. 4 (2019): 1427-1440.

[22] Houben, Sebastian, Johannes Stallkamp, Jan Salmen, Marc Schlipsing, and Christian Igel. "Detection of traffic signs in real-world images: The German Traffic Sign Detection Benchmark." In *The 2013 international joint conference on neural networks (IJCNN)*, pp. 1-8. Ieee, 2013.

[23] Kim, Chang-il, Jinuk Park, Yongju Park, Woojin Jung, and Yong-seok Lim. "Deep learning-based real-time traffic sign recognition system for urban environments." *Infrastructures* 8, no. 2 (2023): 20.

[24] Oruklu, Erdal, Damien Pesty, Joana Neveux, and Jean-Emmanuel Guebey. "Real-time traffic sign detection and recognition for in-car driver assistance systems." In *2012 IEEE 55th International Midwest Symposium on Circuits and Systems (MWSCAS)*, pp. 976-979. IEEE, 2012.
[25] Triki, Nesrine, Mohamed Karray, and Mohamed Ksantini. "A real-time traffic sign recognition method using a new attention-based deep convolutional neural network for smart vehicles." *Applied Sciences* 13, no. 8 (2023): 4793.
[26] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

Machines and Algorithms
http://www.knovell.org/mna

Research Article

Efficiency of K-Prototype and K-Mean algorithm using Support Vector Machine (SVM)

Muhmmad Sharjeel Asad Areeb[1, *] and Nabeel Asghar[1]

[1] Department of Computer Science, Bahauddin Zakariya University, 60000, Multan, Pakistan

[*]Corresponding Author: Muhmmad Sharjeel Asad Areeb. Email: sharjeelasadareeb@gmail.com

Received: 16 December 2024; Revised: 1 February 2025; Accepted: 17 February 2025; Published: 20 March 2025

AID: 004-01-000050

Abstract: Clustering is a key method in unsupervised machine learning, which is commonly used to find latent patterns in unlabeled datasets. This research evaluates the efficacy of K-Means and K-Prototype clustering algorithms using five benchmark datasets that include labeled, unlabeled, and mixed-type data. After routine preprocessing, datasets were divided into 2 to 5 clusters, and a Support Vector Machine (SVM) classifier was used to check the resulting cluster assignments. Experimental results show that K-Means works better on labeled datasets, while K-Prototype works better on unlabeled and mixed-type datasets. Also, accuracy goes down as the number of clusters goes up, and the best results are shown with two clusters. These results show how the type of data and the way the clusters are set up affect how well clustering and classification tasks work.

Keywords: Clustering; Support Vector Machine; K-Prototype; K-Means;

1. Introduction

Unsupervised machine learning is very effective when there isn't much labeled data, it's too expensive, or it's not available [1]. Unsupervised approaches try to find hidden patterns, structures, or relationships in data that hasn't been labeled [2], [3]. This is different from supervised learning, which uses labeled datasets. Clustering is one of the most common methods in this group. It puts related data points into groups based on their traits [4]. Finding natural groupings by clustering is useful in many areas, including healthcare [5], marketing [6], finance [7], image processing [8], and cybersecurity [9].

K-Means is one of the most popular clustering algorithms due to its simplicity, efficiency, and ability to handle large datasets [10], [11]. It works well for numerical data by grouping points so that each cluster has minimal internal variation [12]. However, real-world datasets often contain both numerical and categorical attributes. In such cases, the K-Prototype algorithm is more suitable [1], [13], as it extends K-Means to handle mixed-type data using a different measure of dissimilarity for categorical values [14].

Evaluating clustering performance is challenging because, unlike supervised learning, there are no predefined labels to compare against [15]. One way to address this is by using a post-clustering classification approach [16], [17]. Here, the clusters formed are tested using a supervised model—such as a Support Vector Machine (SVM)—to check how well the data points can be separated based on the clusters [18], [19]. This hybrid method provides an indirect measure of clustering quality and has been explored in previous works [20], [21].

In this study, we compare the performance of K-Means and K-Prototype on both labeled and unlabeled datasets [1], [3]. We test their effectiveness under different conditions, including varying the number of

clusters, and use multiple publicly available datasets [22]. Data preprocessing techniques are applied to reduce noise and improve quality before clustering [23]. We then use SVM to evaluate the separability of the clusters [18].

The paper is organized as follows: Section 2 reviews related work on clustering methods and evaluation techniques. Section 3 describes the datasets and methodology, including preprocessing, clustering, and classification steps. Section 4 presents the experimental setup, results, and comparison between K-Means and K-Prototype on labeled and unlabeled data. Section 5 summarizes key findings and suggests directions for future research in unsupervised learning and clustering evaluation.

2. Literature Review

Unsupervised machine learning, especially clustering, has been widely used in healthcare, cybersecurity, finance, and behavioral analytics because it can find hidden patterns in datasets that don't have labels. K-Means is still one of the most common clustering methods since it works well with big sets of numbers and is easy to scale [1]. However, real-world datasets frequently encompass both numerical and categorical variables, constraining the direct use of K-Means. To solve this problem, Huang [2] came up with the K-Prototype algorithm, which uses the Euclidean distance metric from K-Means for numerical characteristics and the dissimilarity measure from K-Modes for categorical attributes.

Sharma et al. [3] utilized K-Means clustering on patient medical records in healthcare applications to discern high-risk groups, obtaining enhanced prediction performance when combined with Support Vector Machine (SVM) classification. In a similar way, Singla and Bhatia [4] showed that K-Means followed by SVM classification made it much easier to accurately forecast disease categories than clustering alone. These results indicate that post-clustering classification can function as an efficient indirect assessment of clustering quality.

In cybersecurity, Aljawarneh et al. [5] put forward a hybrid intrusion detection model that used K-Means to group network traffic at first and then SVM to classify it, which led to better detection accuracy. Elngar et al. [6] also used a K-Means–SVM pipeline to find anomalies in IoT environments and said that it had lower false positive rates than other methods.

Beyond numerical datasets, Joshi and Dang [7] applied K-Means with SVM classification to predict online user preferences in e-commerce, showing enhanced personalization accuracy. Kumar et al. [8] extended this approach to educational data mining, where clustering was used to identify learning behavior patterns prior to classification.

While prior works have explored K-Means extensively, fewer studies have investigated K-Prototype in conjunction with SVM for mixed-type data. Kumari and Yadav [9] conducted a comparative analysis of K-Means, K-Modes, and K-Prototype, concluding that K-Prototype produced better clustering quality for mixed-attribute datasets. However, they did not evaluate the post-clustering classification performance, leaving a research gap in understanding how such algorithms impact separability in supervised learning. A consolidated summary of related works is presented in **Table 1**.

Table 1: Literature Review Analysis

Ref.	Algorithm(s)	Purpose	Performance	Evaluation Measure
[1]	K-Means, K-Prototype	Performance analysis for outlier detection	Produced better clusters using proposed architecture	Efficiency and accuracy
[2]	Gray Wolf Optimization (GWO), Support Vector Machine (SVM)	Compare accuracy with other methods	Achieved a 27.68% accuracy improvement over baseline methods	Accuracy

[3]	K-Prototype (combination of K-Means and K-Modes)	Measure user behavior	Clusters revealed more accurate user preferences	Clustering of mixed-attribute data
[4]	K-Means	Application in data mining and pattern recognition	Demonstrated efficiency and promising results	Algorithm behavior and performance
[5]	SVM with Sequential Minimal Optimization (SMO)	Improve SVM efficiency	SMO outperformed standard SVM	Classification accuracy
[6]	K-Means + SVM, Weighted SVM (WSVM)	Measure predictive performance using boosting	Both K-Means SVM and WSVM improved classification accuracy	Accuracy comparison
[8]	K-Means, Genetic Algorithm (GA), SVM	Optimal feature selection in data mining	Achieved 98.79% accuracy on reduced datasets	Accuracy of reduced datasets
[9]	K-Means + SVM Classifier (K-SVM)	Hyperplane separation between two classes	Selected most informative samples, improving efficiency	Time efficiency
[7]	K-Means, SVM	Intrusion detection	Achieved 90% detection accuracy	Attack detection accuracy
[10]	Basic K-Means, Enhanced K-Means	Address limitations of K-Means	Enhanced K-Means outperformed Basic K-Means	Efficiency
[11]	Improved K-Means (based on largest minimum distance)	Reduce dependence on initial points and avoid local minima	Improved K-Means outperformed Basic K-Means	Efficiency and time
[12]	K-Means	Improve time efficiency	Removed limitations of standard K-Means	Time efficiency and performance
[13]	K-Means, Euclidean Distance	Data analysis	Provided efficient clustering results	Performance evaluation
[14]	K-Means, SVM	Compare clustering and classification for categorical data	K-Means produced better results	High-dimensional feature space analysis
[15]	SVM, Derivative-Free Numerical Optimizer	Process optimization	Efficiently performed classification tasks	Efficiency
[16]	SVM	Pattern recognition, regression, and operator inversion	Achieved 22× faster results compared to baseline	Accuracy and speed

3. METHODOLOGY

The proposed framework employs a hybrid evaluation strategy that integrates unsupervised clustering with supervised classification to assess the performance of K-Means and K-Prototypes algorithms. The methodology is organized into four primary stages: (1) dataset selection and preprocessing, (2) clustering, (3) supervised classification, and (4) evaluation. The overall workflow is illustrated in **Figure. 1**.

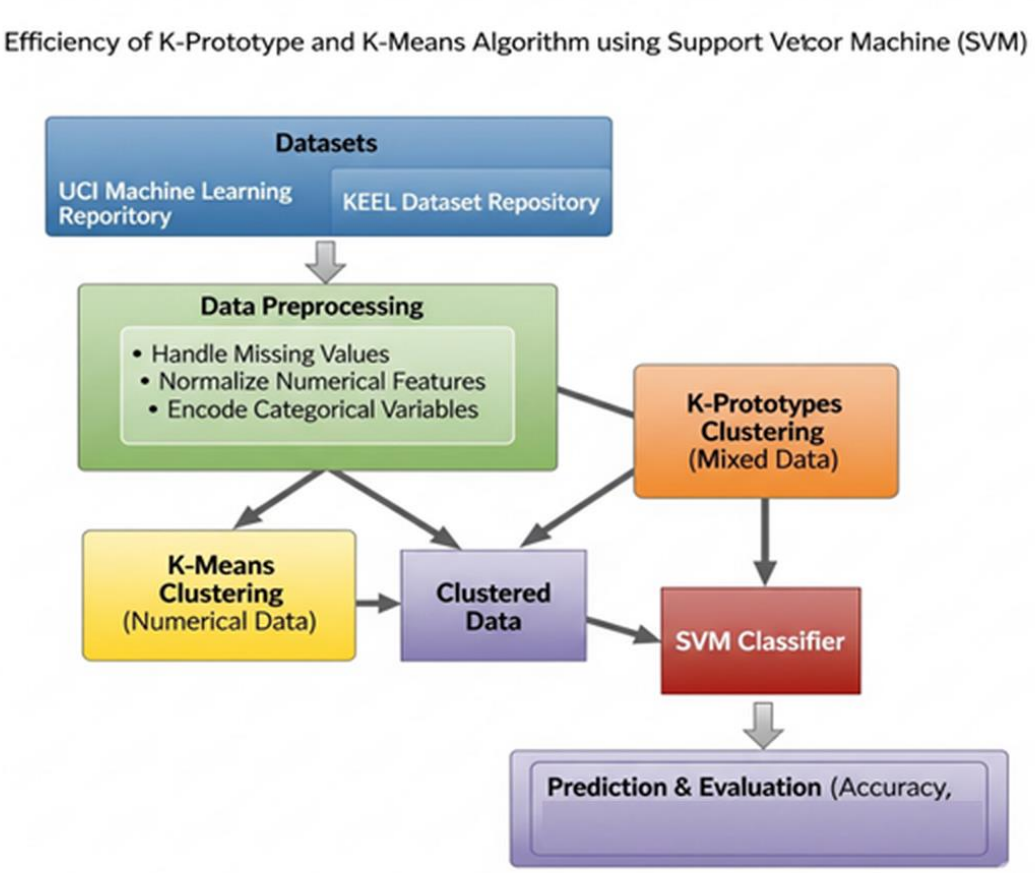


Figure 1: Proposed Methodology Diagram

3.1. Dataset Selection

Eight benchmark datasets were sourced from two widely recognized repositories: the UCI Machine Learning Repository [24] and the KEEL (Knowledge Extraction based on Evolutionary Learning) Dataset Repository [25]. These repositories were selected due to their dataset diversity, established use in prior clustering and classification studies, and suitability for evaluating hybrid algorithms. The datasets encompass both numerical and categorical features, making them appropriate for mixed-type clustering algorithms such as K-Prototypes [1], [3], which combine Euclidean and categorical dissimilarity measures.

The datasets used include: the **Adultt dataset** [24], containing 48,842 instances with 14 categorical and integer attributes, used for income classification with some missing values; the **Hepatitis dataset** [24] with 155 instances and 19 mixed-type attributes for survival prediction, also containing missing data; the **Primary Tumor dataset** [24] comprising 339 categorical attributes aimed at tumor location classification with missing values; the **Arrhythmia dataset** [24] with 452 instances and 279 mostly real-valued attributes for arrhythmia diagnosis, containing missing data; the **Monks-Problems dataset** [24] with 432 categorical instances used for binary classification, having no missing values; the **Airlines Delay dataset** [25] with 500 instances and categorical/integer features for flight status classification, without missing values; the **Credit Approval dataset** [24] containing 690 mixed-type attributes for credit approval classification, including missing values; and the **Vowel (Japanese Vowels) dataset** [24], a time series dataset with 640 real-valued attributes used for classification, without missing values. These datasets provide a

comprehensive test bed for evaluating clustering algorithm performance across varied data types and domains. A summary of the selected datasets, including size, type, and missing values, is presented in Table 2.

Table 2: Summary of Selected Datasets

Dataset Name	# Instances	# Attributes	Data Types	Purpose	Missing Values
Adultt [24]	48,842	14	Categorical, Integer	Classification	Yes
Hepatitis [24]	155	19	Real, Categorical, Integer	Classification	Yes
Primary Tumor [24]	339	17	Categorical	Classification	Yes
Arrhythmia [24]	452	279	Real, Categorical, Integer	Classification	Yes
Monks-Problems [24]	432	7	Categorical	Classification	No
Airlines Delay [25]	500	7	Categorical, Integer	Classification	No
Credit Approval [24]	690	15	Real, Categorical, Integer	Classification	Yes
Vowel (Japanese) [24]	640	12	Real	Classification	No

3.2. Data Preprocessing

To ensure compatibility with the clustering algorithms and enhance performance, each dataset underwent a standardized preprocessing pipeline.

- Missing values were addressed using mean or mode imputation [26], depending on the attribute type, while records with excessive incompleteness were removed to maintain data integrity.
- Numerical features were normalized to a [0,1] range using Min–Max scaling [27] to prevent bias in distance-based clustering.
- For categorical variables, preprocessing differed based on the clustering algorithm: in the case of K-Means [2], categorical attributes were transformed into numerical form through one-hot encoding [28], whereas for K-Prototypes, categorical attributes were preserved in their native form to fully leverage the algorithm's capability to handle mixed-type data.

This ensured comparability between the two clustering approaches and prevented distance bias.

3.3. Model Architecture

SMO classifier was used to evaluate two clustering algorithms: K-Means and K-Prototypes. Five datasets were tested both with and without the class attribute to compare classification accuracy. Initially, the K-Means technique was applied to each dataset, varying the number of clusters from 2 to 5. After clustering, the resulting grouped datasets—first including the class attribute and then with the class attribute removed—were fed into the SMO classifier to measure accuracy. Subsequently, the K-Prototypes clustering algorithm was applied under the same conditions to assess its performance with and without class labels.

3.4. Clustering Phase

In the initial stage, datasets were grouped without the use of class labels to identify inherent data structures.

- For purely numerical datasets, the K-Means algorithm [2] was employed, utilizing Euclidean distance as the similarity measure.
- In contrast, for mixed-type datasets containing both numerical and categorical attributes, the K-Prototypes algorithm [1], [3] was applied, combining Euclidean distance for numerical attributes with simple matching dissimilarity for categorical attributes.

This clustering process revealed the underlying patterns within the data, forming the foundation for subsequent supervised evaluation. In this study, the number of clusters was systematically adjusted from 2 to 5 to assess the impact of cluster granularity on classification accuracy. This range was chosen because it strikes a compromise between simplicity (fewer clusters) and granularity (more clusters), and it doesn't go too far with the number of clusters, which can lead to over-segmentation. It is important to note that no automatic cluster number discovery approach, such the elbow method or silhouette analysis, was used. Instead, the goal was to look at how clustering worked and how well the classifier worked with a set range of cluster values.

3.5. Classification Phase

To quantitatively evaluate the quality of clustering, the resulting cluster assignments were mapped to the actual class labels using a Support Vector Machine (SVM) classifier [29].

This supervised step was included for two reasons:

1. **Validation:** checking how well unlabeled clusters align with ground truth via majority voting [30].
2. **Predictive testing:** assessing if discovered clusters improve downstream classification accuracy. It facilitated predictive performance assessment, leveraging the SVM's ability to model both linear and non-linear decision boundaries [31] to achieve robust classification, even in high-dimensional and complex datasets.

3.6. Performance Evaluation

To evaluate the performance of trained SVM classifier over clustered data, we have exploited 'Accuracy' classification metric. The formulae for its evaluation have been mentioned below.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

4. Results and Discussion

The performance of K-Means and K-Prototype clustering algorithms was evaluated across multiple datasets using the SMO classifier for accuracy measurement. The datasets used included Adultt, Hepatitis, Primary Tumor, Arrhythmia, Monks-Problems, Airlines, Credit, and Vowel datasets. For each dataset, clustering was performed with the number of clusters ranging from two to five, and experiments were conducted using both datasets containing the class attribute and those with the class attribute removed. To figure out how well the K-Means and K-Prototype clustering algorithms worked, we used the SMO classifier in a 10-fold cross-validation scenario to find the classification accuracy. This method makes sure that the results aren't biased toward either the training or the test data because each dataset is split into training and testing subsets in several folds. The accuracies that were provided are the average performance across all folds, not just the training data.

For the **K-Means** clustering technique, the Adultt dataset, comprising 48,842 instances and 14 attributes, showed the highest classification accuracy of 97.57% when clustered into two groups including the class attribute. However, at three and four clusters, the model trained on data without the class attribute outperformed that with class labels. Similarly, the Hepatitis dataset, with 155 instances and 19 attributes, achieved its peak accuracy of 92.85% at two clusters with class information included. Interestingly, for

clusters of four and five, the classifier trained on data excluding the class attribute demonstrated better performance. The Primary Tumor dataset (339 instances, 17 categorical attributes) achieved perfect accuracy (100%) at two clusters for both data variants, and also at three clusters without the class attribute.

The Arrhythmia dataset, which is more complex with 452 instances and 279 attributes, attained 100% accuracy at two clusters for both dataset types. For higher cluster counts, models trained without class labels showed improved results compared to those with class information. Finally, the Monks-Problems dataset, containing 432 instances and seven categorical fields, reached a maximum accuracy of 92.3% with two clusters across both dataset types. At higher cluster counts (3 to 5), models trained without the class attribute outperformed those with it. Overall, the K-Means results consistently indicated that classification accuracy declines as the number of clusters increases, with two clusters providing optimal accuracy. Furthermore, inclusion of the class attribute generally benefitted models at low cluster counts, whereas for higher cluster counts, models trained on data without class labels sometimes achieved better performance.

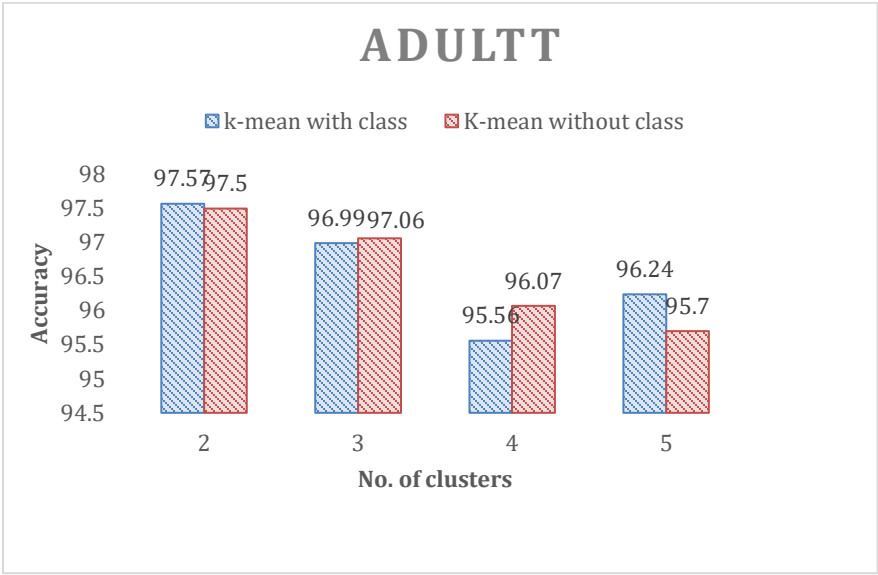


Figure 2: Performance of K-Means on ADULTT Dataset

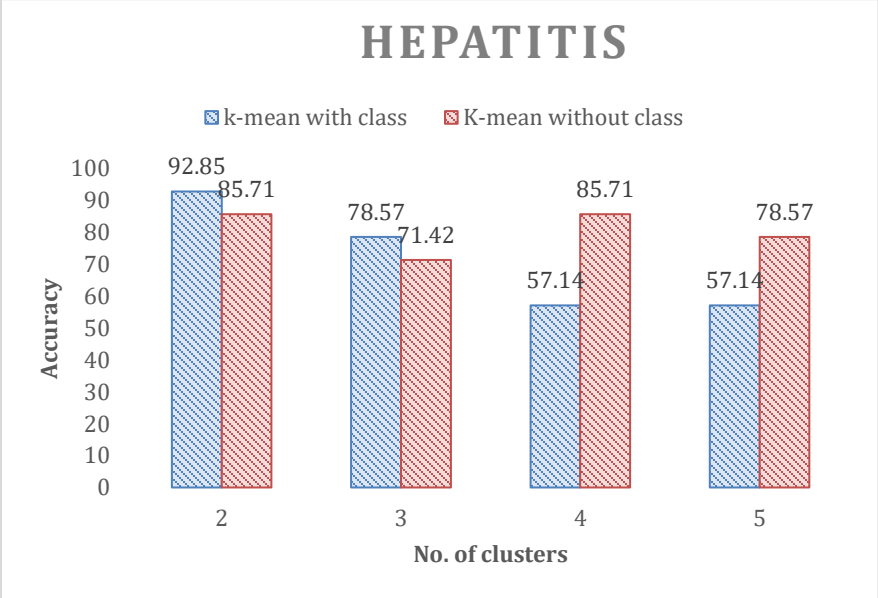


Figure 3: Performance of K-Means on HEPATITIS Dataset

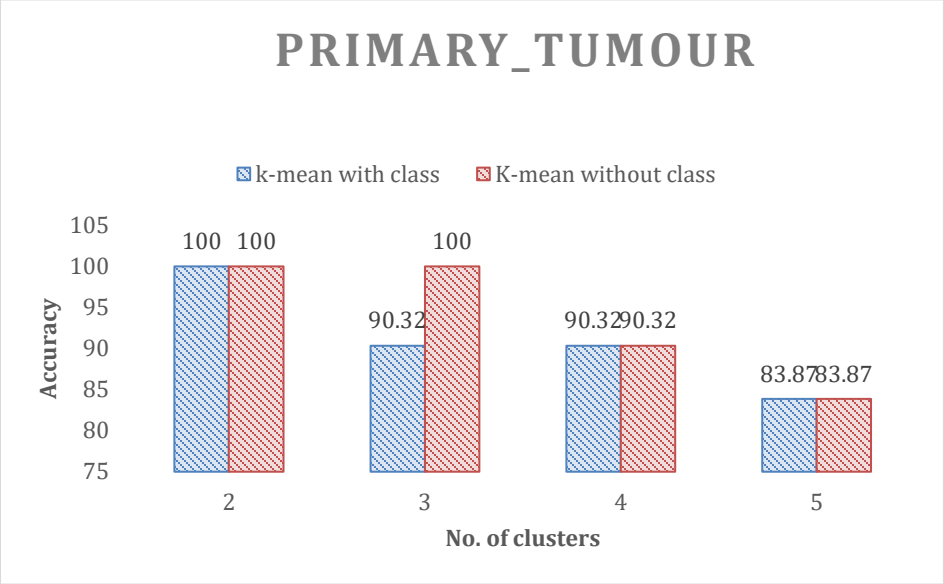


Figure 4: Performance of K-Means on PRIMARY_TUMOUR Dataset

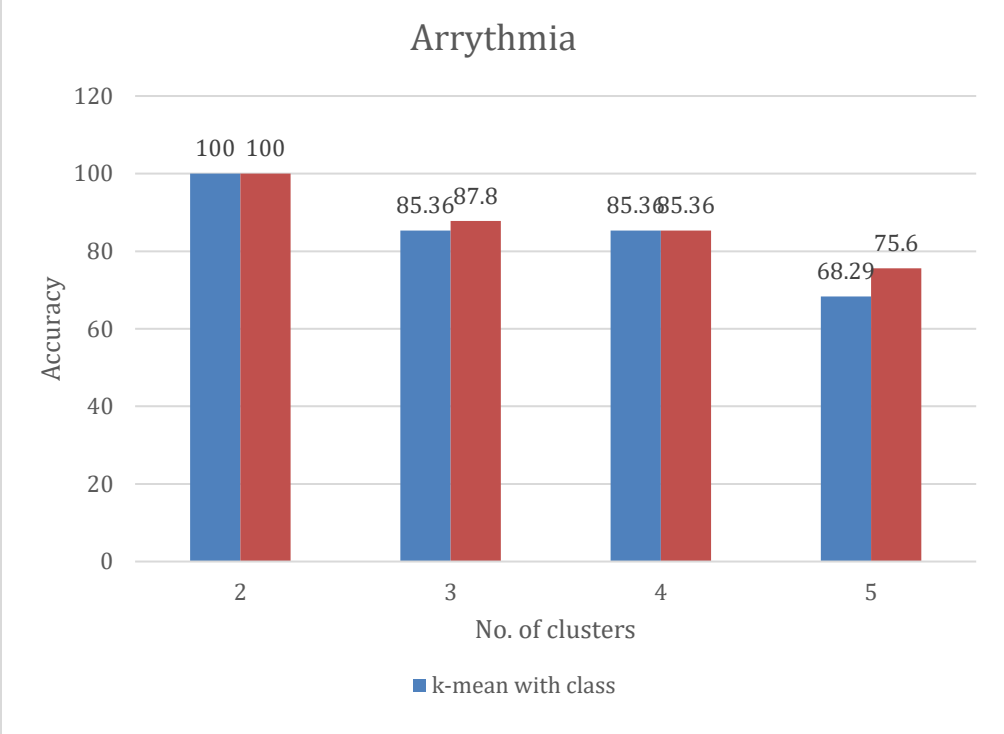


Figure 5: Performance of K-Means on ARRYTHMIA Dataset

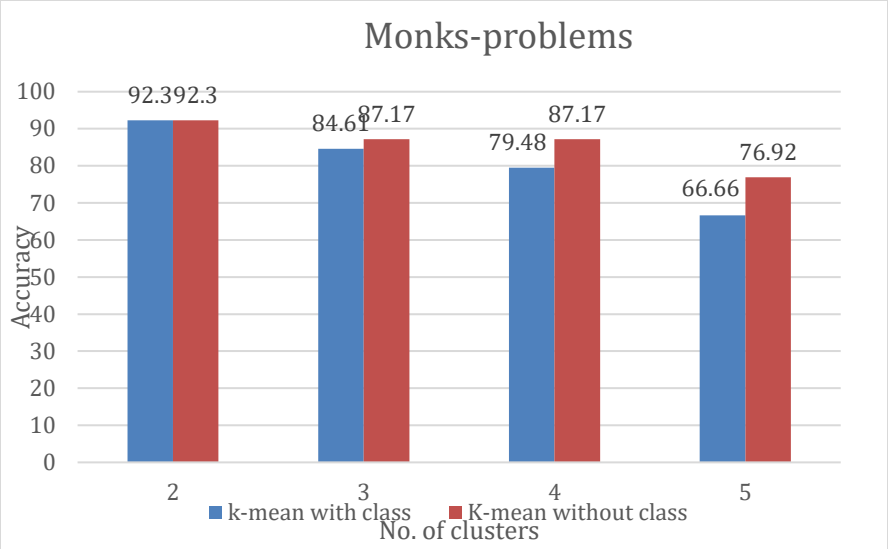


Figure 6: Performance of K-Means on MONKS-PROBLEM Dataset

The **K-Prototype** clustering algorithm showed a similar trend of decreasing accuracy with increasing cluster counts. On the Adultt dataset, the highest accuracy of 99.24% was achieved with two clusters on data without the class attribute, surpassing K-Means results. For three to five clusters, models trained on datasets including the class attribute performed better. The Airlines dataset, comprising 500 instances and seven attributes, attained its peak accuracy of 98.73% at two clusters, equally on datasets with and without class labels. The Credit dataset (690 instances, 15 attributes) achieved maximum accuracy of 96.33% with two clusters on both dataset types; however, at three clusters, models trained with class labels outperformed, while at four and five clusters, models without class labels were superior. The Hepatitis dataset yielded 97.82% accuracy with two clusters on data containing the class attribute. Lastly, the Vowel dataset, a time-series dataset with 640 instances and 12 features, showed maximum accuracy of 96.96% at two and three clusters for data without the class attribute. Models trained with class labels performed better at two and three clusters, while models without class labels excelled at higher cluster counts. These results suggest that the K-Prototype algorithm often outperforms K-Means, especially when class labels are removed, likely due to its suitability for mixed data types.

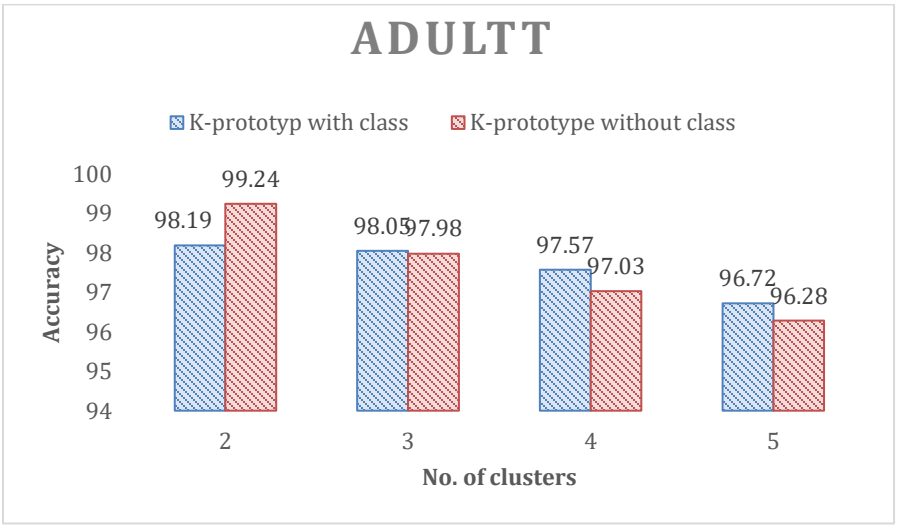


Figure 7: Performance of K-Prototype on ADULTT Dataset

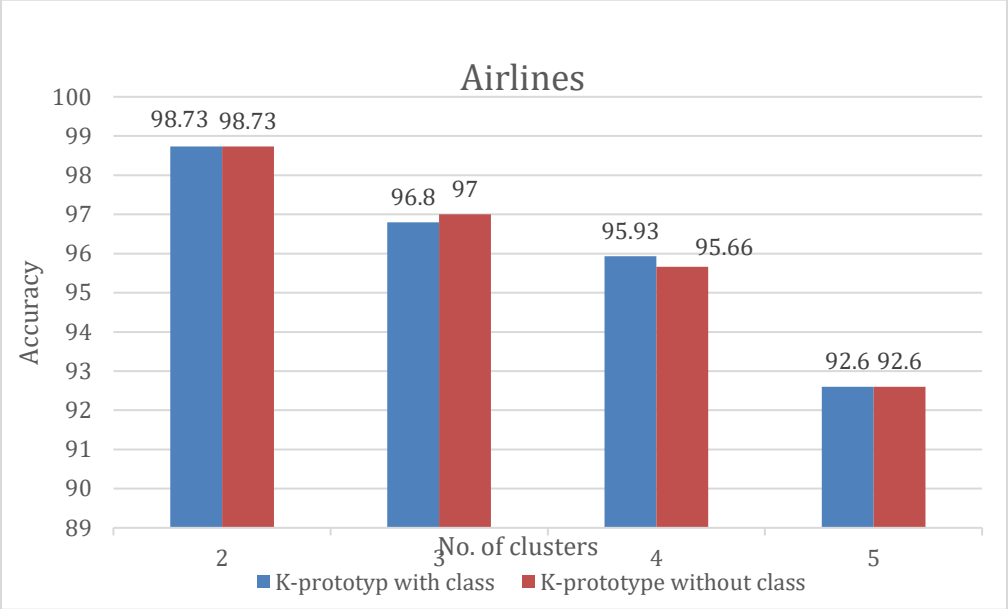


Figure 8: Performance of K-Prototype on AIRLINES Dataset

Figure 9: Performance of K-Prototype on CREDIT Dataset

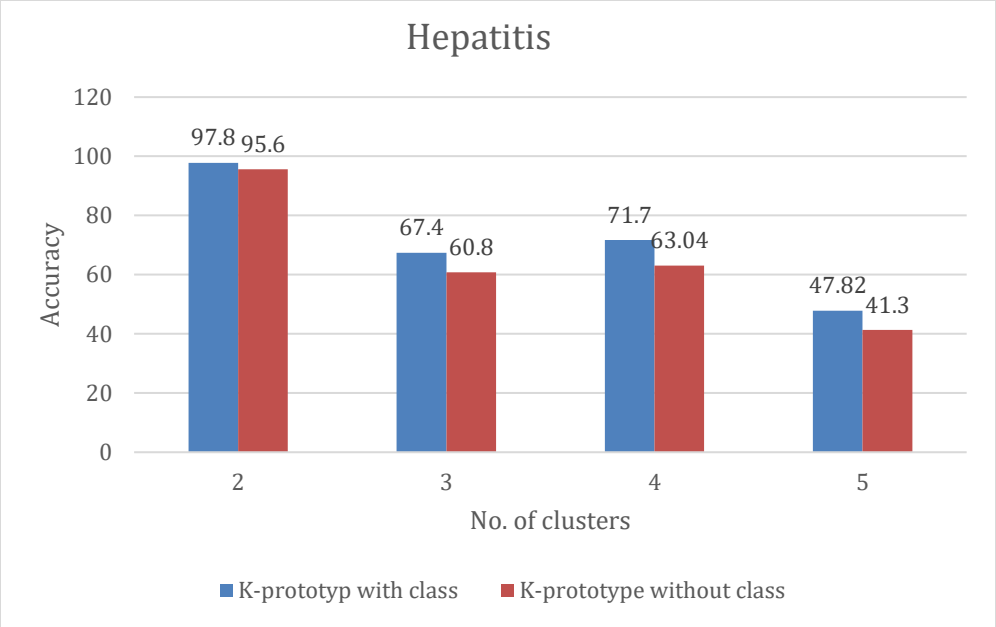


Figure 10: Performance of K-Prototype on HEPATITS Dataset

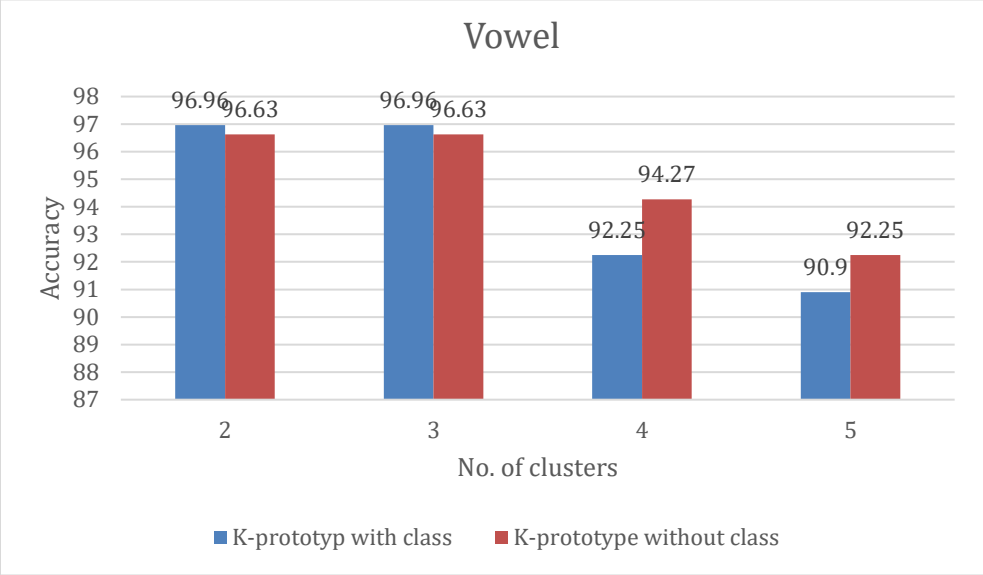


Figure 11: Performance of K-Prototype on VOWEL Dataset

Table 3 summarizes the best accuracy results achieved across all datasets for both algorithms. It is evident that both algorithms perform optimally with two clusters, and that the presence or absence of class attributes influences classifier performance differently depending on the dataset and clustering method. The general decline in accuracy with increasing cluster count may indicate over-segmentation, which negatively impacts cluster homogeneity and classifier performance.

Table 3: Results Evaluation

Dataset	Algorithm	Best Cluster Count	Accuracy with Class (%)	Accuracy without Class (%)	Majority Best
Adultt	K-Means	2	97.57	–	With Class
Hepatitis	K-Means	2	92.85	–	With Class
Primary Tumor	K-Means	2, 3	100	100	Without Class (at 3)
Arrhythmia	K-Means	2	100	100	Without Class
Monks-Problems	K-Means	2	92.3	92.3	Without Class
Adultt	K-Prototype	2	~95.28 (at 5 clusters)	99.24	Without Class
Airlines	K-Prototype	2	98.73	98.73	Equal
Credit	K-Prototype	2	96.33	96.33	Without Class
Hepatitis	K-Prototype	2	97.82	–	With Class
Vowel	K-Prototype	2, 3	~96.96	96.96	Without Class

The difference in performance between K-Means and K-Prototypes can be explained by the types of datasets and the algorithms themselves. K-Means only uses Euclidean distance, which makes it better for datasets with mostly numerical features (like Arrhythmia and Monks-Problems) but not as good for datasets with a mix of types (like Adultt or Credit Approval). K-Prototypes, on the other hand, combines numerical and categorical differences, which makes it better at finding patterns in datasets that include a lot of different types of data. This is why it works better on datasets like Adultt and Airlines, where categorical variables are very important for categorization.

The number of clusters is another thing that affects accuracy. The best results for both algorithms were at two clusters. As the number of clusters increased, the accuracy went down. This drop could be because of over-segmentation, which happens when you break data into smaller groups. This makes clusters less homogeneous and makes it harder for the SVM classifier to map clusters back to ground-truth labels. It is interesting that models trained without the class property occasionally did better than those trained with labels. This implies that in some instances, eliminating the class attribute mitigated bias and enabled clustering algorithms to identify more organic groupings, which were later confirmed using supervised classification.

In general, the comparison results show that K-Means is easier to compute and works well with numerical datasets, whereas K-Prototypes is more versatile and works better with mixed-type datasets in the real world. This supports the idea of using a hybrid framework that uses supervised learning to check the quality of clustering, making sure that unsupervised results are not only statistically sound but also useful for making predictions.

This study mainly looks at K-Means and K-Prototypes, but it is also necessary to include other sophisticated clustering approaches that have become popular in recent years. DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is great at finding noise spots and dealing with clusters of any shape, but it needs careful adjustment of its parameters and has trouble with changing densities. Gaussian Mixture Models (GMM) presume that data comes from a mix of Gaussian distributions. This lets

clusters take on different shapes, but GMMs are sensitive to how they are set up and need to know how many components they have. Spectral Clustering uses graph theory to group non-convex structures, but it is quite expensive to do this on big datasets.

K-Means is still fast and useful for purely numerical data, and K-Prototypes makes this efficiency work for mixed-type data as well. This means that both algorithms are good candidates for large-scale real-world datasets.

5. Conclusion and Future work

In this study, we investigated the performance of two prominent unsupervised clustering algorithms, K-Means and K-Prototype, across multiple benchmark datasets. Our experimental framework involved conducting clustering with varying numbers of clusters (from 2 to 5) and evaluating classifier accuracy using the SMO (SVM) classifier on datasets both with and without the class attribute. The results consistently showed that both clustering methods achieve their best performance when the number of clusters is set to two. Increasing the number of clusters usually made the classifier less accurate, which could mean that the clusters were too small and not strong enough to hold together. For K-Means, classifiers trained on datasets containing the class attribute usually did better with fewer clusters, while models trained without class labels usually did better with more clusters. On the other hand, the K-Prototype approach usually gave more accurate results when trained on datasets that didn't have the class property. This shows that it is good at working with heterogeneous data types. In general, the performance trends of both algorithms were similar for cluster counts greater than two.

Our results show how important it is to choose the right number of clusters and the right data set for clustering-based classification to work well. These insights enhance comprehension of the efficient application of unsupervised clustering algorithms across various data sources and classification challenges.

In future research, we intend to expand this study by examining a wider array of clustering algorithms, encompassing hierarchical, density-based, and model-based approaches. Also, using ensemble clustering methods and more advanced feature selection methods could make clustering quality and classification accuracy even better. Another good idea is to look at how different data pretreatment methods and dimensionality reduction methods affect the results of clustering. In the end, these efforts are meant to provide a stronger and more flexible clustering framework that can handle complicated real-world datasets.

Funding Statement: No grants have been received from any funding bodies to conduct this study.

Conflicts of Interest: The authors declare the absence of conflicts of interest.

Data Availability: All datasets exploited in this study are taken from open-source online data repositories.

References

[1] Ahmad, Izhar. "K-mean and K-prototype algorithms performance analysis." *International Journal of Computer and Information Technology* 3, no. 04 (2014): 823-828.

[2] Kamel, Seyed Reza, Reyhaneh YaghoubZadeh, and Maryam Kheirabadi. "Improving the performance of support-vector machine by selecting the best features by Gray Wolf algorithm to increase the accuracy of diagnosis of breast cancer." *Journal of Big Data* 6, no. 1 (2019): 90.

[3] Ranti, Kiefer Stefano, Kelvin Salim, and Abba Suganda Girsang. "Clustering Steam User Behavior Data using K-Prototypes Algorithm." In *Journal of Physics: Conference Series*, vol. 1367, no. 1, p. 012018. IOP Publishing, 2019.

[4] Ali, Huda Hamdan, and Lubna Emad Kadhum. "K-means clustering algorithm applications in data mining and pattern recognition." *International Journal of Science and Research (IJSR)* 6, no. 8 (2017): 1577-1584.

[5] Wen, Zeyi, Bin Li, Ramamohanarao Kotagiri, Jian Chen, Yawen Chen, and Rui Zhang. "Improving efficiency of SVM k-fold cross-validation by alpha seeding." In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31, no. 1. 2017.

[6] Kim, SungHwan. "Weighted K-means support vector machine for cancer prediction." *Springerplus* 5, no. 1 (2016): 1162.
[7] Shrivastava, Alka, and Ram Ratan Ahirwal. "A SVM and K-means clustering based fast and efficient intrusion detection system." *International Journal of Computer Applications* 72, no. 6 (2013).
[8] Santhanam, T., and M. S. Padmavathi. "Application of K-means and genetic algorithms for dimension reduction by integrating SVM for diabetes diagnosis." *Procedia Computer Science* 47 (2015): 76-83.
[9] Yao, Yukai, Yang Liu, Yongqing Yu, Hong Xu, Weiming Lv, Zhao Li, and Xiaoyun Chen. "K-SVM: An Effective SVM Algorithm Based on K-means Clustering." *J. Comput.* 8, no. 10 (2013): 2632-2639.
[10] Singh, S. P., and Asmita Yadav. "Study of k-means and enhanced k-means clustering algorithm." *International Journal of Advanced Research in Computer Science* 4, no. 10 (2013): 103-107.
[11] Li, Youguo, and Haiyan Wu. "A clustering method based on K-means algorithm." *Physics Procedia* 25 (2012): 1104-1109.
[12] Chakraborty, Sanjay, and Naresh Kumar Nagwani. "Analysis and study of incremental k-means clustering algorithm." In *International Conference on High Performance Architecture and Grid Computing*, pp. 338-341. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.
[13] Oyelade, Olanrewaju Jelili, Olufunke O. Oladipupo, and Ibidun Christiana Obagbuwa. "Application of k Means Clustering algorithm for prediction of Students Academic Performance." *arXiv preprint arXiv:1002.2425* (2010).
[14] Marino, Marina, and Cristina Tortora. "A comparison between K-means and Support Vector Clustering for Categorical Data." *Statistica applicata* 21, no. 1 (2009): 5-16.
[15] Eitrich, Tatjana, and Bruno Lang. "Efficient optimization of support vector machine learning parameters for unbalanced datasets." *Journal of computational and applied mathematics* 196, no. 2 (2006): 425-436.
[16] Burges, Christopher J., and Bernhard Schölkopf. "Improving the accuracy and speed of support vector machines." *Advances in neural information processing systems* 9 (1996).
[17] Kuswardana, Dendy Arizki, Dwi Arman Prasetya, Trimono Trimono, and I. Gede Susrama Mas Diyasa. "Comparison of Elbow and Silhouette Methods in Optimizing K-Prototype Clustering for Customer Transactions." *Jurnal Ilmiah Edutic: Pendidikan dan Informatika* 12, no. 1 (2025): 43-48.
[18] Aschenbruck, Rabea, Gero Szepannek, and Adalbert FX Wilhelm. "Initialization strategies for clustering mixed-type data with the k-prototypes algorithm: R. Aschenbruck et al." *Advances in Data Analysis and Classification* (2025): 1-30.
[19] Ping, Yuan, Huina Li, Chun Guo, and Bin Hao. "k ProtoClust: Towards Adaptive k-Prototype Clustering without Known k." *Computers, Materials & Continua* 82, no. 3 (2025).
[20] Alrasheed, Mousa, and Monjur Mourshed. "Building stock modelling using k-prototype: A framework for representative archetype development." *Energy and Buildings* 311 (2024): 114111.
[21] Mohd, Azimah, Lay Eng Teoh, and Hooi Ling Khoo. "Passengers' requests clustering with k-prototype algorithm for the first-mile and last-mile (FMLM) shared-ride taxi service." *Multimodal Transportation* 3, no. 2 (2024): 100132.
[22] Shi, Yan, Siyuan Zhang, Siwen Wang, Hui Xie, and Jianying Feng. "Multiple-perspective consumer segmentation using improved weighted Fuzzy k-prototypes clustering and swarm intelligence algorithm for fresh apricot market." *Italian Journal of Food Science* 36, no. 4 (2024): 38.
[23] Jing, Xin, and Hao Gao. "An Improved K-PROTOTYPE Clustering Algorithm and Its Application." In *Proceedings of the 2023 6th International Conference on Machine Learning and Natural Language Processing*, pp. 182-186. 2023.