Research Article

Qinling He* and Wei Zhang

Algorithm selection model based on fuzzy multi-criteria decision in big data information mining

https://doi.org/10.1515/dema-2023-0156
received July 12, 2023; accepted January 20, 2024

Abstract: In the era of big data, efficient classification of rapidly growing data volumes is a critical challenge. Traditional algorithms often fall short in handling the scale and complexity of big data, leading to inefficiencies in classification accuracy and processing times. This study aims to address these limitations by introducing a novel approach to algorithm selection, which is essential for advancing big data classification methods. We developed an advanced classification algorithm that integrates a fuzzy multi-criteria decision-making (MCDM) model, specifically tailored for big data environments. This integration involves leveraging the analytical strengths of MCDM, particularly the analytic hierarchy process, to systematically evaluate and select the most suitable classification algorithms. Our method uniquely combines the precision of fuzzy logic with the comprehensive evaluative capabilities of MCDM, setting it apart from conventional approaches. The proposed model is meticulously designed to assess key performance indicators such as accuracy, true rate, and processing efficiency in various big data scenarios. Our findings reveal that the proposed model significantly enhances classification accuracy and processing efficiency compared to traditional algorithms. The model demonstrated a marked improvement in true rates and overall classification performance, showcasing its effectiveness in handling large-scale data challenges. These results underline the model's potential as a pragmatic solution for big data classification, offering substantial improvements over existing methodologies. The study contributes a groundbreaking perspective to the field of big data classification, addressing critical gaps in current practices. By combining fuzzy logic with MCDM, the proposed model offers a more nuanced and effective approach to algorithm selection, catering to the intricate demands of big data environments. This research not only enhances the understanding of classification behaviors in big data but also paves the way for future advancements in data mining technologies. Its implications extend beyond theoretical value, providing practical tools for practitioners and researchers in the realm of big data analytics.

Keywords: information mining, *K*-means, fuzzy multi-attribute decision-making, fuzzy logic in big data, algorithm efficiency in data mining, big data scalability

MSC 2020: 68T05, 90B50

1 Introduction

The advent of the big data era, propelled by rapid advancements in science and technology, has elevated the significance of data classification across diverse sectors, including finance, healthcare, and education. In this increasingly data-driven world, the ability to classify data accurately and expediently is not just a technical

*** Corresponding author: Qinling He,** School of Education, Glasgow University, Glasgow, G128QQ, United Kingdom, e-mail: heeeeed0330yf@163.com
Wei Zhang: School of Marxism, Yangtze University, Jingzhou, China, e-mail: zwcjdx123456@163.com

task but a crucial element in enhancing decision-making efficiency and reducing error rates. Data classification, a pivotal aspect of data mining, enables the transformation of raw data into categorized, actionable insights. This process facilitates various functions such as data prediction, risk mitigation, impact analysis, and model formulation, thereby playing a vital role in informing and guiding strategic decisions. The surge in the volume and complexity of data, however, presents new challenges and opportunities in the field of data classification. With the proliferation of big data classification algorithms, the task of selecting the most suitable algorithm from a multitude of options has emerged as a key concern in data mining. This issue is not only technical but also strategic, as the choice of an algorithm can significantly impact the accuracy and speed of data processing, ultimately affecting the quality of insights derived from the data. This evolving landscape has captured the attention of scholars and practitioners worldwide, leading to a surge in research aimed at developing, refining, and evaluating classification algorithms suitable for big data environments. The quest to address this challenge has opened up new frontiers in data mining, with a focus on creating algorithms that are not only efficient in handling large volumes of data but also capable of adapting to the dynamic and often unstructured nature of big data. The exploration of these algorithms is crucial for harnessing the full potential of big data, turning vast data streams into meaningful information that can drive innovation and progress across various domains.

In the current landscape of big data analysis, the selection of appropriate classification algorithms is not just a matter of computational efficiency, but also of decision-making under complex, multi-dimensional criteria. This is where multi-criteria decision-making (MCDM) models become pivotal. MCDM offers a robust framework for evaluating a multitude of conflicting and diverse criteria, which is a common scenario in big data classification tasks. The complexity of big data necessitates the consideration of various factors such as scalability, accuracy, processing speed, and adaptability to data diversity. MCDM models adeptly handle these multifaceted criteria, providing a systematic approach to decision-making.

Among the various MCDM methodologies, the analytic hierarchy process (AHP) stands out, particularly in the context of this study. AHP, developed by Thomas L. Saaty in the 1970s, is renowned for its ability to simplify complex decision-making processes by breaking them down into a hierarchy of more manageable sub-problems, each of which can be analyzed independently. In our study, AHP is used to dissect the intricate process of algorithm selection into several levels of criteria and sub-criteria, thereby enabling a thorough and nuanced evaluation of potential algorithms.

The application of AHP in this study is significant for several reasons. First, AHP facilitates the inclusion and balancing of both qualitative and quantitative aspects of algorithm performance. This is crucial in big data scenarios where certain qualitative factors, such as algorithm robustness or adaptability to varying data structures, play a key role. Second, AHP allows for the prioritization of these criteria based on their relevance and impact, which is essential in tailoring the algorithm selection to specific big data tasks. This prioritization is achieved through pairwise comparisons and the subsequent generation of a weighted scoring system, offering a transparent and methodical approach to decision-making.

Moreover, the integration of AHP with fuzzy logic in our model addresses the inherent uncertainties and vagueness present in big data environments. Fuzzy logic complements AHP by providing a mathematical structure to capture the imprecision inherent in human judgments. This combination enhances the model's ability to handle real-world scenarios where data are often incomplete, ambiguous, or subject to rapid changes.

Although many researchers have made some achievements in the field of classification algorithm, it is still an important challenge to select the suitable algorithm from the numerous algorithms in the big data environment. In addition, it is also difficult to test whether the operation process of classification algorithm is reliable and efficient.

In order to solve the aforementioned problems, the following technical methods and strategies are adopted in this study: (1) a classification algorithm optimization model based on fuzzy multi-criteria decision is proposed and verified in the big data environment. The model comprehensively considers the performance indicators of various classification algorithms, such as accuracy rate, true rate, true-negative rate, recall rate, and accuracy rate, and it evaluates and sorts different algorithms by fuzzy multi-criteria decision-making (MCDM) method, so as to provide users with a more reliable basis for algorithm selection. (2) The parallel

design of five alternative classification algorithms is realized on the Hadoop big data platform. In this study, CPAR algorithm, C4.5 algorithm, neural network algorithm, Bayesian network algorithm, and *K*-means clustering algorithm are selected as alternative algorithms, and they are parallelized. The experimental results show that these five algorithms all achieve good performance in the big data environment. (3) Fuzzy MCDM method is applied to the evaluation of big data classification algorithms. By standardizing the parallel operation results of five alternative classification algorithms, a decision matrix suitable for the evaluation model of big data classification algorithms is formed. Then, the evaluation model of big data classification algorithm based on the fuzzy MCDM method evaluates and sorts the five algorithms, which provides users with more targeted algorithm selection suggestions.

The contributions of this research in advancing the field of big data classification through the development of an integrated fuzzy MCDM model and an advanced classification algorithm represent a significant leap in addressing the inherent challenges of big data analytics. The novel approach adopted in our study goes beyond traditional algorithmic enhancements, offering profound insights and impactful advancements in several key aspects of data science. By weaving together the intricate elements of fuzzy logic with the structured approach of MCDM, our model emerges as a paradigm shift in algorithm selection, tailored for the multifaceted nature of big data. It embodies a significant progression from conventional methods, bringing a nuanced understanding of complex data structures and the dynamism of big data environments. This integration not only enhances the accuracy and efficiency of classification tasks but also enriches the decision-making process with a multi-dimensional perspective, acknowledging and addressing the uncertainties and variabilities inherent in large datasets.

Furthermore, our study's contributions transcend the theoretical realm, offering tangible tools and methodologies that can be readily applied in diverse practical scenarios. The adaptability and scalability of our model make it an invaluable asset in various industries grappling with the challenges of big data, from healthcare and finance to e-commerce and beyond. By enabling a more nuanced and efficient handling of data classification tasks, our research paves the way for more sophisticated data analytics, leading to better-informed business decisions, more accurate predictive models, and a deeper understanding of complex data patterns.

Moreover, the insights provided by our research foster a conducive environment for future explorations and innovations in the field. By challenging the traditional paradigms and introducing a novel approach, we set a new course for subsequent studies to build upon, potentially leading to breakthroughs in algorithm design, data processing, and analytical methodologies. The holistic perspective offered by our model, encompassing both theoretical robustness and practical applicability, marks a significant advancement in the field of big data analytics, offering a comprehensive solution to some of the most pressing challenges in the domain.

In this manuscript, we present a comprehensive exploration of algorithm selection for big data classification, featuring the innovation of a fuzzy MCDM model. Section 1 sets the stage by introducing the critical challenges inherent in big data environments, such as scale, complexity, and real-time processing demands, underscoring the limitations of traditional classification methodologies. Section 2 dives into the heart of our research, detailing the development of our novel fuzzy MCDM model. Here, we integrate the precision of fuzzy logic with the analytical depth of MCDM, focusing on the AHP to create a robust framework for algorithm selection. Section 3 presents an extensive comparative analysis, where our model is evaluated against established methods, including the RANCOM (RANking COMparison) method. This section highlights the superior performance of our approach in terms of accuracy, efficiency, scalability, and adaptability through rigorous experimental validations using augmented datasets. In Section 4, we explore the innovative aspects and practical applications of our model. This includes its utility in real-world big data scenarios, its capability to address existing gaps in classification methodologies, and implications across various industries. Section 5 concludes the manuscript, offering a thoughtful discussion on future directions for big data classification. We highlight potential enhancements to our model, explore its applications in emerging technological domains, and outline its role in advancing the field of data analytics. Through this manuscript, we provide a detailed and holistic view of the complexities and solutions in big data classification, contributing significantly to the academic and practical realms of data science.

2 Related research

Data mining plays an increasingly important role in today's society, among which classification is one of the key tasks in data mining. Many classification algorithms have been proposed, such as *K*-mean clustering algorithm, neural network algorithm, C4.5 algorithm, and CPAR algorithm, which are widely used to solve problems in the big data environment. However, traditional classification model evaluation methods mainly focus on accuracy and running time and lack evaluation of other specific data indicators, such as true rate, true-negative rate, and accuracy rate. In order to better evaluate the merits and demerits of classification model under certain conditions, it is necessary to adopt more systematic and detailed evaluation criteria. Fuzzy MCDM method, as a common evaluation method, can describe the fuzziness in MCDM better and is beneficial to evaluate the data in multiple directions. Therefore, it is of great significance to evaluate the classification model using the fuzzy MCDM method.

The central theme of this article is the groundbreaking application of fuzzy sets and their generalizations within the realms of MCDM and AHP, with a specific focus on big data classification. This innovative approach is designed to bridge a critical gap in existing data science methodologies, which often struggle to effectively manage the complexity and ambiguity present in large datasets. This study emphasizes the transformative potential of this approach in revolutionizing big data analytics by offering classification models that are more nuanced, adaptable, and efficient.

Delving into the research findings that form the backbone of this focus, this article draws on Çalı and Balaman's (2019) work, which underscores the role of fuzzy sets in enhancing decision-making processes, particularly in complex big data scenarios. Their research is pivotal in illustrating the necessity of incorporating fuzzy logic within MCDM frameworks to adeptly handle ambiguous and incomplete data. Similarly [1], Yang et al. (2020) explored the advancements in fuzzy MCDM methods and their significant impact on the evaluation of classification algorithms in big data contexts, where traditional evaluation metrics often prove inadequate [2].

Further exploration by Farzin et al. [3] and Jiang et al. [4] sheds light on the implications of fuzzy sets in enhancing both accuracy and reliability in big data analysis, especially in sectors such as healthcare and finance that are dynamic and data-intensive. Complementing this perspective are the studies by Lamrini et al. [5], Djenadic et al. [6], and Mohaghegh et al. [7], which reinforce the importance of fuzzy logic in big data analytics, addressing challenges related to scalability and real-time processing.

The practical utility of fuzzy MCDM models is emphasized through the work of Ziemba et al. [8], Ge et al. [9], and Wang et al. [10], which demonstrate the models' adaptability across a range of industries and diverse data structures. Lu et al. [11], Meng et al. [12], and Masdari and Khezri [13] provided further insights into the role of fuzzy logic in optimizing computational efficiency and managing multi-lingual and multi-modal data, aspects critical to the evolving needs of big data environments.

In addition, the application of fuzzy sets in environmental and risk assessment models, as illustrated by Meng et al. [14] and Rafiei Sardooi et al. [15], reveals the expansive applicability of this approach in various domains. By concentrating on the integration of fuzzy sets in MCDM and AHP for big data classification, this article makes a substantial contribution to the understanding of managing large, complex datasets. It paves the way for the development of more accurate, efficient, and scalable data analysis techniques, responding adeptly to the increasing demands of the data-driven world, thus marking a significant advancement in the field of big data analytics.

2.1 Fuzzy set theory

Generally speaking, fuzzy sets were proposed by Lotfi A. Zadeh in 1965 and have a development history of several decades. This method takes the object to be examined and the fuzzy concepts reflecting it as a certain fuzzy set, and establishes appropriate membership functions. It analyzes fuzzy objects through operations and transformations related to fuzzy sets. Fuzzy set theory is based on fuzzy mathematics and studies phenomena related to non precision. Fuzzy set theory has been applied in many aspects (Ohlan and Ohlan [16]; Tian et al. [17]).

In recent years, scholars have made a series of progress in the evaluation of fuzzy set methods and classification algorithms. For example, Intuitionistic fuzzy sets are proposed to further extend fuzzy set theory (Ali et al. [18]; Xue and Deng [19]; Alkan and Kahraman [20]). Fuzzy-set qualitative comparative analysis (fsQCA) is becoming increasingly prevalent in management research and other ereas (Kumar et al. [21]).

2.2 MCDM

MCDM or multi-criteria decision analysis is a subdiscipline of operation research that explicitly evaluates multiple conflicting guidelines in the decision-making process, whether in everyday life or in settings such as business, government, and medicine. MCDM refers to a way of making decisions in which a centralized screening is performed in some conflicting scenarios that cannot be shared. These options can be finitely many or infinitely many. MCDM is one of the important research contents in the field of decision-making. It has the following characteristics:

(1) There are a large number of items in MCDM. It can evaluate, judge and queue many projects, and finally select the most ideal project as the final goal.
(2) MCDM refines the decision factors. In the process of project research, the MCDM considers the impact factor of each project as the criterion affecting the final result of the project, and the value of each impact factor must be filtered, screened, and processed to extract effective information and assign weight to the factor according to the different importance of the impact factor.
(3) MCDM adopts the way of multi-dimensional decision-making. In general, MCDM will treat the number of items as a matrix, which is used for judgment and decision. Then, several to a dozen decision methods are used by the multi-criteria decision to organize information. This can lead to a dynamic analysis system with very powerful judgment mechanisms.

MCDM method can use modern information technology to process information quickly, process effective information, and increase execution efficiency. This kind of decision-making method can react quickly according to the requirements of decision-makers, which has a powerful role in helping decision-makers.

The application of fuzzy logic in big data analytics, especially in the recent half-decade, has been transformative. Its ability to manage the ambiguities and uncertainties characteristic of vast and complex datasets has made it indispensable in the field. This period has seen a notable shift toward integrating fuzzy logic with advanced machine learning techniques, thereby enhancing the interpretability and accuracy of models applied to large-scale data. Such integration has proven particularly effective in scenarios such as internet of things (IoT) data analysis and cloud computing, where data dynamism and unstructured formats present unique challenges.

One of the most salient trends in this domain has been the application of fuzzy logic to optimize resource allocation and data storage in cloud-based big data platforms. This approach not only improves efficiency but also addresses critical security concerns, making it a cornerstone of modern data management strategies. In parallel, the role of fuzzy logic in promoting sustainable and green computing practices within big data centers has gained attention. Here, the focus has been on optimizing energy consumption and enhancing the environmental sustainability of data processing.

In the social networking sphere, fuzzy sets have been instrumental in analyzing large volumes of data to understand complex user behavior and sentiments. This application is particularly notable given the explosion of social media usage and the consequent generation of massive unstructured datasets. Furthermore, the integration of fuzzy logic with blockchain technology has opened new vistas in securing big data systems, ensuring data integrity and privacy in an increasingly decentralized digital landscape.

The healthcare sector has also witnessed the burgeoning use of fuzzy MCDM, where making accurate diagnostic and treatment decisions based on extensive data has become more efficient and reliable. Additionally, the development of adaptive algorithms for streaming big data, applying fuzzy logic principles, has marked a significant stride in managing the rapid flow and changing nature of data in real time.

Another critical area of advancement is the creation of hybrid models that amalgamate fuzzy logic with deep learning techniques. This synergy has brought forth models with enhanced robustness and interpretability, which is crucial for complex big data applications.

In synthesizing these developments, this study is uniquely positioned at the intersection of these emerging trends. The specific focus of our research is the application and generalization of fuzzy sets within MCDM and AHP frameworks to refine big data classification algorithms. This focus is not merely an academic pursuit but is driven by the imperative to develop robust, flexible, and accurate tools for big data analysis. Our study recognizes and addresses the critical need for accommodating the uncertainty and subjectivity in data – factors often overlooked yet essential in the analytics of large and complex datasets.

2.3 Fuzzy MCDM

In recent years, fuzzy analysis and fuzzy decision models and related concepts have been established. Fuzzy MCDM can play a role in uncertainty problems. It is found that fuzzy set theory is very suitable to describe the fuzziness in MCDM. Therefore, the research on the combination of MCDM and fuzzy number has become an important research direction and hotspot of fuzzy MCDM methods.

The decision-making process of fuzzy multi-criteria decision is usually composed of two parts: one is to determine the attribute weight and fuzzy index value. After determining these two values, a reasonable fuzzy operator is selected and the two values are normalized to combine them into a fuzzy utility value. The other is to form a set of fuzzy utility values, use appropriate sorting method to compare and sort the set, and finally select the optimal scheme according to the sorting results and return it to the decision-maker.

The weight determination methods in fuzzy multi-criteria decision include subjective weight determination method and objective weight determination method. In this article, the objective weight determination method is mainly used. The objective weighting method includes the following main features.

(1) It does not depend on the subjective position of the decision-maker and emphasizes the differences among the indicators to be evaluated in the assessment object;
(2) High transparency in the evaluation process;
(3) The weight of indicators is not inherited; in different stages, if the value of the evaluation index changes, the weight factor of each index will change;
(4) Calculations are generally based on more complete mathematical theories, especially in improved theories where calculations are usually more complex.

2.4 Research progress of classification algorithms

2.4.1 Algorithm introduction

2.4.1.1 Decision tree algorithm

Decision tree algorithm is a top-down construction process, starting from the root node to divide the feature attributes until the category of leaf nodes is determined. The construction process of decision tree involves many steps such as feature selection, tree generation, and pruning. Zikopoulos and Eaton (2011) proposed ID3 algorithm, which is a decision tree generation algorithm based on information gain. In the feature selection of ID3 algorithm, information gain is used as the partitioning standard [22]. The larger the information gain is, the greater the “purity” improvement obtained by using this feature for partitioning. However, ID3 algorithm has some limitations, such as poor processing of continuous value attributes, cannot process missing values, etc. Quinlan (1993) also proposed C4.5 algorithm, which improved on ID3, introduced information gain rate as the partitioning standard, and could deal with continuous value attributes and missing values [23].

2.4.1.2 Neural networks

Neural network simulates the structure and function of biological nervous system and is a nonlinear statistical data modeling tool. The neural network is mainly composed of input layer, hidden layer, and output layer, in

which each neuron has connection weight and activation function. In the 1940s, the concept of neural network was proposed, but its research has been stagnant. It was not until the 1980s that neural network research began to flourish with the advent of backpropagation algorithms. Backpropagation algorithm is a method to adjust the weight by minimizing the prediction error, so that the neural network can achieve better performance when solving complex problems. Neural networks are widely used, including image recognition, speech recognition, natural language processing, and other fields.

2.4.1.3 Bayesian network algorithm

Bayesian networks are an approach based on a probability graph model that is used to represent probabilistic relationships between variables. Bayesian networks are represented as directed acyclic graphs, where nodes represent the random variables, edges represent the causal relationships between variables, and conditional probability tables represent the conditional probability relationships between nodes. Bayesian networks have good interpretability and reasoning ability and are suitable for expressing and analyzing uncertain and potential events, especially when dealing with incomplete, imprecise, and fuzzy information.

2.4.1.4 KNN algorithm

KNN algorithm is a learning method based on instances. By calculating the distance between the data to be classified and the data in the training set, it finds the nearest K neighbors and then votes according to the labels of these neighbors to get the category of the data to be classified.

Table 1: Comparison of advantages and disadvantages of each algorithm

Algorithm	Advantages	Disadvantages
Decision tree algorithm	a. Easy to understand and explain good visualization effect b. Strong ability to deal with multiple classification problems c. Insensitive to missing values and able to process irrelevant feature data	a. Sensitive to noise data and easy to overfit b. Low computational efficiency when dealing with large-scale data c. The construction process of decision trees may be unstable, and subtle changes in data may lead to significant changes in tree structure
Neural network	a. Suitable for complex and nonlinear problems b. With strong approximation ability, can learn any continuous function c. Can realize distributed storage and processing, with a certain degree of fault tolerance	a. The model is complex, and the training process may be slow b. It is difficult to interpret and understand what is learned c. Prone to falling into local optimal solutions
Bayesian network algorithm	a. Good performance in the case of sparse or incomplete data b. Can handle uncertainty and potential events c. Models can be easily updated to accommodate new data	a. High computational complexity, especially when dealing with high-dimensional data b. Strong assumption of data independence, which may not be the case in reality c. A large amount of training data is required
KNN classification algorithm	a. The algorithm is simple and easy to implement b. It has good classification effect for nonlinear data c. Suitable for multi-classification problems	a. Large amount of computation, especially on large datasets b. Sensitive to noise and irrelevant features c. Need to choose the right K value
CPAR algorithm	a. Combines the advantages of association rules and classification algorithms b. Use greedy algorithm to create rules, avoiding redundancy c. Dynamic methods avoid double calculations when creating rules	a. It is possible to generate too many rules b. Sensitive to noise in the data c. Reliance on dynamic methods may affect prediction accuracy

2.4.1.5 CPAR algorithm

CPAR algorithm is a classification algorithm based on association rules. CPAR algorithm combines association rule mining technology with classification task, and it realizes classification by mining predictive association rules in training data. To a certain extent, CPAR algorithm solves the difficulties of traditional classification algorithms in processing large-scale and high-dimensional data and has high classification accuracy (Table 1).

2.4.2 Comparison of advantages and disadvantages of the algorithm

Compared with other classification algorithms, the KNN algorithm has the following advantages (Table 2).

However, KNN algorithm also has disadvantages, such as large computation, affected by outliers, and sensitive to unbalanced data. Therefore, the following content will focus on the research progress of the KNN algorithm-derivative algorithm.

Table 2: Advantages of KNN algorithm compared with other algorithms

Decision tree algorithm	No feature selection: KNN algorithm directly uses the distance between data for classification, without the need for feature selection High tolerance for noisy data: KNN algorithm determines the classification result by voting of multiple neighbors, which can reduce the influence of noisy data
Neural network algorithm	Easy to understand and implement: KNN algorithm is simpler and easier to understand and implement than neural network algorithm No training: KNN algorithm needs no training process, while neural network algorithm needs weight adjustment and training, which consumes time and computing resources
Bayesian network algorithm	No assumptions about data distribution: KNN algorithms make no assumptions about data distribution, whereas Bayesian network algorithms are usually based on conditional probability and Bayesian formulas and require assumptions about data distribution Suitable for multi-classification problems: KNN algorithm is suitable for multi-classification problems, while Bayes network algorithm may need to construct multiple binary classification models when dealing with multi-classification problems
CPAR algorithm	Simple algorithm: KNN algorithm is relatively simple and intuitive, easy to understand and implement No training: KNN algorithm is an instance-based learning method, no training process, save time and computing resources Multi-classification problems: KNN algorithm is suitable for multi-classification problems, while CPAR algorithm may need to construct multiple binary classification models when dealing with multi-classification problems

2.5 *K*-means algorithm

2.5.1 Basic principles of *K*-means algorithm

K-means clustering algorithm is an iterative solution of the clustering analysis algorithm; the step is to divide the data into k groups in advance, then randomly select k objects as the initial clustering center, and then calculate the distance between each object and each seed clustering center. Assign each object to the closest clustering center from it. The cluster center and the objects assigned to it represent a cluster. For each sample assigned, the cluster center of the cluster is recalculated based on the existing objects in the cluster. This process is repeated until a certain termination condition is met. The termination condition can be that the number (or minimum number) of objects is reassigned to a different cluster, number (or minimum number) of cluster centers changes again, and the sum of squares of error is locally minimum. Its algorithm implementation steps are as follows:

(1) Feature selection: k samples are randomly selected as the mean vector of the initial cluster class.
(2) Sample partitioning: each sample dataset is divided into the closest cluster to it.
(3) Update: update the mean vector of the cluster class according to the cluster to which each sample belongs.
(4) Repeat Step 2 and Step 3. When the set number of iterations is reached or the mean vector of cluster class is no longer changed, the model construction is completed and the clustering algorithm results are output.

2.5.2 Advantages and disadvantages of *K*-means algorithm

K-means algorithm has the following advantages: it is iterative and can overcome the inaccuracy of clustering of a small number of samples; it can optimize the unreasonable classification of initial supervised learning samples; aiming at some small samples can reduce the total clustering time complexity. However, *k*-means algorithm also has some disadvantages: the *K* value needs to be set in advance, which belongs to the prior knowledge; *K*-means algorithm is very sensitive to the initial clustering center; the algorithm does not work for all data types (Table 3).

Table 3: Nearest neighbor vs *K*-means comparison

KNN	***K*-means**
1. KNN is the sorting algorithm 2. Supervised learning 3. The dataset fed to it is labeled data, which is already completely correct data	1. *K*-means is the clustering algorithm 2. Unsupervised learning 3. The dataset fed to it is unlabeled data, which is disorganized and becomes somewhat sequential only after clustering, first disordered and then ordered
There is no obvious early training process, which belongs to memory-based learning	There is an obvious early training process
The meaning of *K*: A sample *x* is given, and to classify it, i.e., to find its y, find the nearest *K* data points near *x* from the dataset. Among these *K* data points, category C occupies the most number, so set the label of *x* as *C*	Meaning of *K*: *K* is a manually fixed number. Assuming that the dataset can be divided into *K* clusters, it requires a little prior knowledge because it is manually fixed
Similarities: Both involve the process of, given a point, finding the nearest point in the dataset, i.e., both use an algorithm called NN (nearest neighbor), which is typically implemented using KD trees.	

2.5.3 Nearest neighbor versus *K*-means

Fuzzy *K*-means algorithm: Fuzzy *K*-means algorithm is derived from the *K*-means algorithm. In the process of clustering, although the results obtained each time are not necessarily the expected effect, the boundaries between categories are clear, and the clustering center is modified according to the samples currently available in each category. In the process of clustering, the category boundaries obtained by the fuzzy *K*-means algorithm are still fuzzy every time, and all samples are needed to modify the clustering center of each class. In addition, the clustering criteria also reflect the fuzziness. The main steps of FK-means are as follows (Abdullah 2013):
(1) Determine the number of clusters.
(2) Initialize the membership matrix U.
(3) Update the cluster center according to the membership matrix.
(4) Determine whether the difference of the objective function is less than threshold C. If not, iteratively solve the membership degree according to the clustering center and return to Step 3.

3 Hadoop-distributed parallel computing framework

In this section, we will introduce the Hadoop-distributed parallel computing framework and distributed parallel *K*-means algorithm in the computing framework.

3.1 Hadoop introduction

The Hadoop-distributed framework can store very large datasets, and in the process of data processing, the data are divided and stored in the cluster sub-nodes. This framework has good scalability, fault tolerance, and reliability, and the interface management page is simple and easy to operate (Figure 1).

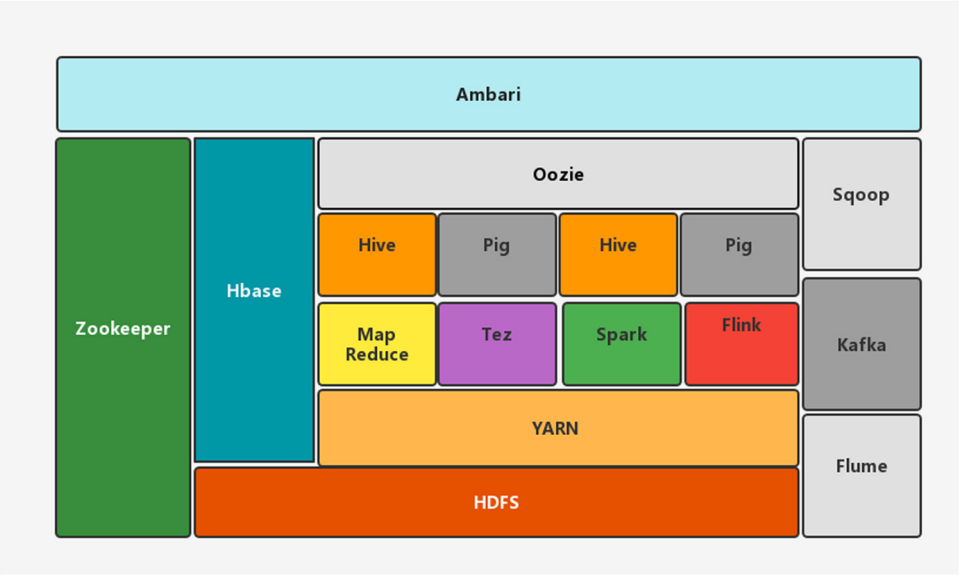

Figure 1: Structure of the Hadoop ecosystem.

3.2 Hadoop architecture

The Hadoop distributed file system (HDFS) is the lowest layer of the Hadoop ecosystem. It can store a large amount of structured data and provide computing resources at the upper layer. The MapReduce-distributed parallel computing framework distributes computing tasks among multiple nodes and writes them to disks for storage. The HBase-distributed database provides high performance, reliability, and scalability. It can write data in batches to achieve data storage and parallel computing. Zookeeper manages data and is widely used in various clusters. Flume can collect logs and realize data preprocessing.

The traditional data storage and processing system has been unable to meet the needs of big data. HBase module is particularly important in the Hadoop ecosystem. It uses distributed storage mode for storage, realizes innovation of storage mode, queries rules by definition, realizes automatic database index, and increases the column storage data. If the data are empty, it will not be stored, saving disk space. To achieve column field data storage, so as to improve query efficiency, HBase can automatically partition data and implement high-concurrency read and write. This feature can satisfy the processing of traffic flow data, reduce the types of traffic data dynamically, and fully display the data characteristics, so as to realize real-time data query.

3.3 Architecture design of the system

The system designed in this article is based on the Hadoop-distributed architecture. The core of the Hadoop framework is HDFS and MapReduce, which can realize the underlying data storage, and MapReduce can realize the massive data calculation. The main characteristics of Hadoop are reliability, efficiency, and

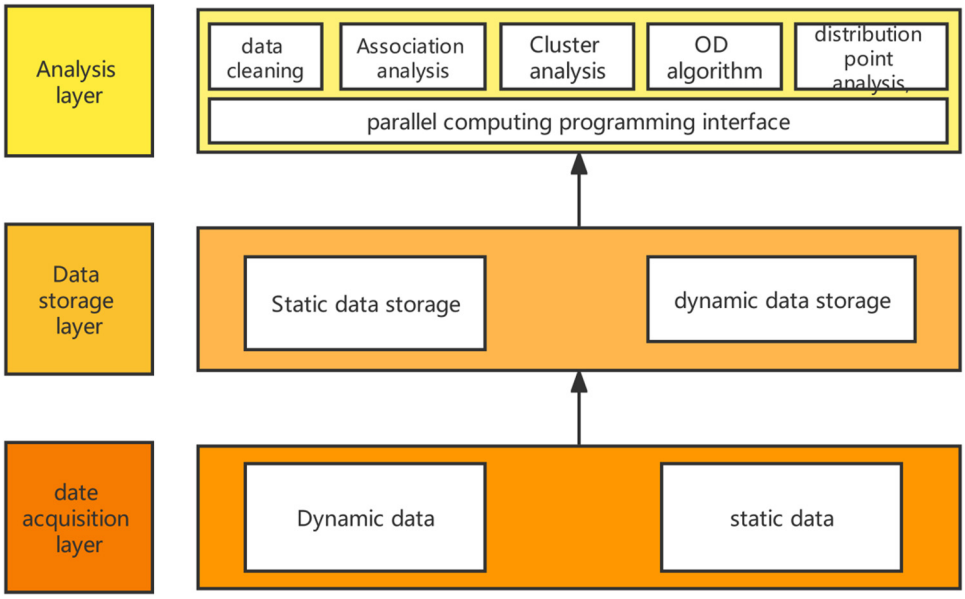

Figure 2: Architecture of the system.

scalability. Based on the characteristics of big data, the parallel processing cloud system is designed. Figure 2 shows the structure of the system. The overall architecture of the system includes the acquisition layer, the analysis layer, and the storage layer. The data acquisition layer includes static and dynamic data collection. The dynamic data is stored in distributed database after integration, and the static data is mapped into RDF data by the later ontology model to realize storage.

The data analysis layer uses MapReduce programming model to realize data calculation and processing based on different analysis requirements of big data, and it uses cluster analysis, association analysis, data cleaning, and other modules in the system. First of all, it is necessary to clean the data and remove the abnormal and unreasonable data. Correlation analysis and cluster analysis are the main analysis methods of data mining. Cluster analysis can use iterative operation to find the appropriate central value, divide the traffic density level, and facilitate reference. The data storage layer enables the collected data to be stored in the Hadoop computer cluster. The computer cluster uses the master-slave architecture. The master node refers to the management node and has a recorded data storage location. In this framework, dynamic data is consolidated into standard data and stored in the Hive data warehouse; static data is mapped to RDF data in an ontology model and then written to an HBase-distributed database.

The calculation results of the data analysis layer are provided to users in the transportation industry through interfaces. Different calculation results are provided according to different requirements of users, and a new computing module is set in the data analysis layer to meet new requirements.

3.4 MapReduce – Parallel computing method based on distributed file system

MapReduce: a software framework for parallel data processing, including the Map function, Reduce function, and main function. The Map function takes a set of data and converts it; the Reduce function computes a new list of keys/values; the main function is to combine job control and file input/output (Figure 3).

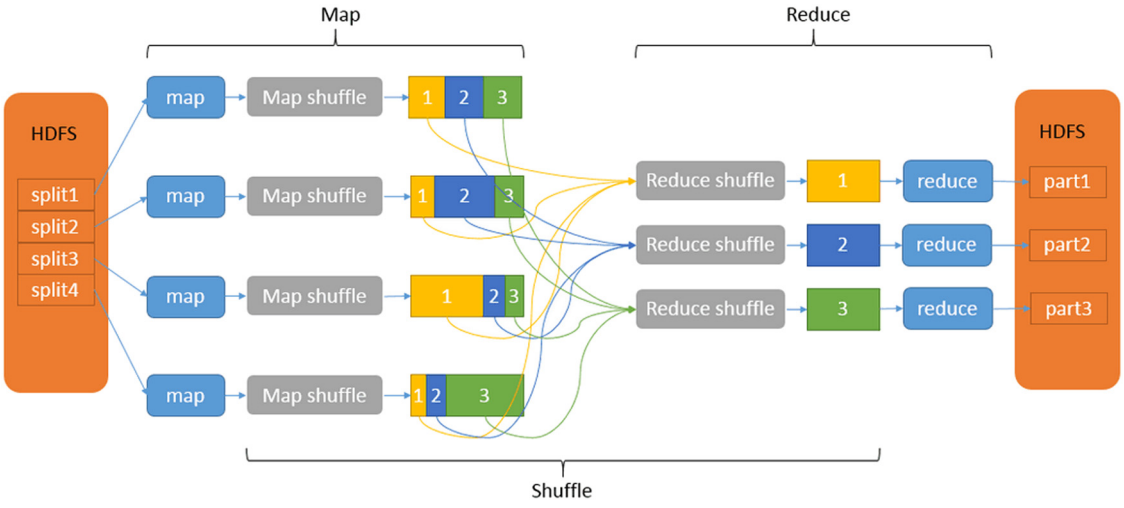

Figure 3: Calculation model of MapReduce.

3.5 *K*-means algorithm under Hadoop-distributed file system

K-means algorithm is a data mining classification algorithm with simple operation and low time complexity. It has been widely used in the data mining of medicine, finance, and other industries, and its process is shown in Figure 4.

In order to realize the good application of *K*-means algorithm in big data processing, the idea of maximum and minimum distance is adopted. The process is as follows:

(1) Suppose there are n objects, as shown in the following formula: $S_n = \{X_1, X_2, \ldots, X_n\}$.
(2) Randomly select an object, for example, for the first type of cluster center, take Euclidean distance as the measurement index of the similarity relationship between data objects, and regard the object with the largest distance between middle and middle as the object; $X_1 S_n X_1 X_2$.
(3) Calculate the distance between the remaining objects and take the minimum as $X_1, X_2 D_x$.

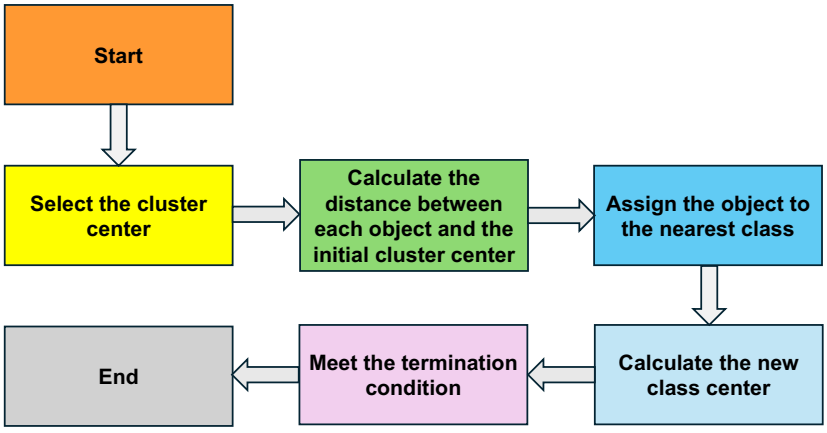

Figure 4: Process of *K*-means algorithm.

(4) Calculate: $\max S_n\{D_N\}$

$$\text{If } \max S_n\{D_X\} > m[\text{average}\,(|X_2 - X_1|)].$$

Then, as the new cluster center, in $X_i 1/2 \leqslant m < 1$;

(5) Repeat Steps 1 through 4 until there are no new cluster centers.

When working with big data, you can start by sampling the data and reducing the size of the data ($0 \leqslant f \leqslant 1$). Assuming that there are n samples in the sample set, class cluster C has at least one sample point. If the sample size is n, it should meet $f|C|$

$$s \geqslant f_n + \frac{n}{|C|}\log\left(\frac{1}{\xi}\right) + \frac{n}{|C|}\sqrt{\left(\log\frac{1}{\xi}\right)^2 + 2f|C|\log\left(\frac{1}{\xi}\right)}.$$

Then, in the sample set, the probability that the number of samples from class cluster C is less than $f|C|\xi(0 \leqslant \xi \leqslant 1)$.

The parallelization implementation of the improved K-means algorithm on Hadoop platform can be divided into three MapReduce processes:

(1) MapReducel: independent parallel sampling via map function. Data is divided into different parts, which reduces the number of tasks. Then, the reduce function clusters the data according to the idea of maximum and minimum distance. Output multiple cluster centers and the average distance of each cluster.
(2) MapReduce2: The output of reduce in Step 1 is processed by map function. Reduce summarizes the output of Step 1, merges adjacent cluster centers, and recalculates the new cluster centers.
(3) MapReduce3: Divide all objects into the nearest cluster by map function and calculate the new cluster center by reduce function until the cluster center is no longer changed. At this point, the calculation of K-means parallel data has been completed according to the aforementioned content.

n: Represents the number of objects or data points being considered in the algorithm. In the context of big data, this can be a large number, indicative of the vast datasets typically processed.

K: This symbol denotes the number of clusters into which the data points are to be grouped. The selection of "K" is crucial as it directly influences the clustering results. In the K-means algorithm, "K" is predetermined and represents a core aspect of the algorithm's functionality.

Euclidean distance: used as a measurement index for the similarity relationship between data objects. It is a common metric in clustering algorithms to determine the "distance" between data points, thereby influencing their grouping.

Cluster center: This term refers to the central point of each cluster. In K-means, cluster centers are recalculated in each iteration to better represent the grouping.

Objective function: denotes the function used to evaluate the performance of the clustering. In K-means, the objective function typically aims to minimize the variance within each cluster.

3.6 Experimental verification and result analysis

The experimental cluster uses six computers, 8-core CPU, 1t hard disk, and 16G memory. Gigabit Ethernet was used. One is na-menode, and the others are Datanodes. The specific construction process of Hadoop platform is as follows:

(1) The Ubuntu 14.04 system is installed on each node in a dual-system mode.
(2) Install the JDK package, then use the ja-va-version command to check if the installation was successful.
(3) Generate the key pair on namenode with the "ssh_keygen-trsan" command, then copy the public key id_rsa.pub into the authorized key file of all nodes to implement the secure Shell protocol (SSH)

configuration. Unzip and install the Ha-doop package, and configure files such as core-site.xml, Ahdfs-site.xml, Amapred-site.xml.

(4) The Hadoop file package is sent to other nodes using the secure copy (SCP) command.
(5) Format the namenode with the "Hadoop Namenode-format" command, then start all processes with the "start-all.sh" command.
(6) After Hadoop is established, write MapReduce function of improved *K*-means algorithm in Java language and run it on the Hadoop platform.

The dataset used in this article is Iris dataset widely used in cluster analysis, including three categories: SetosaA, Versicotor, and Vig-MCA. Due to the small Im capacity, in order to verify the effect of parallel algorithm on big data processing, the original dataset capacity was increased by code, and five datasets of different scales were randomly generated.

In order to verify the performance of *K*-means parallel algorithm, an algorithm evaluation scheme based on fuzzy MCDM method will be made.

In the evaluation process of *K*-means parallel algorithm using fuzzy MCDM method, another thought is used, i.e., to select the algorithm that is most suitable for running the target dataset. The algorithm evaluation model is divided into four parts: target data, alternative algorithm, evaluation index, and evaluation method (Table 4).

Table 4: Experimental dataset

Dataset	Sample number	Number of attributes	Number of categories
Skin_momskin	775,987	43	2
Blood	486,520	26	2

3.7 Experimental results and treatment

In order to facilitate the evaluation of the classification algorithm, the following algorithm evaluation indexes are used as the evaluation criteria of the algorithm, including accuracy, true positive rate (TPR), true negative rate (TNR), false negative rate (FNR), *F* value, and area under curve (AUC), which are expressed as decimal percentages.

The performance of the five classification algorithms running the skin nonskin dataset is shown in Table 5.

The results of five sorting algorithms running the blood dataset are shown in Table 6.

Table 5: Running performance of skin nonskin dataset

Algorithm	Accuracy	TPR	TNR	FNR	AUC	*F* value
CPAR	0.7541	0.8416	0.5134	0.844	0.4214	0.8853
Bayes Net	0.412	0.4225	0.745	0.8454	0.7455	0.8562
C4.5	0.7464	0.8914	0.7561	0.8454	0.9412	0.8744
Neural network	0.4785	0.781	0.8854	0.5453	0.8456	0.5454
k-means	0.8275	0.8208	0.8448	0.8545	0.915	0.7451

Table 6: Results of five algorithms running blood datasets

Algorithms	Accuracy	TPR	TNR	FNR	AUC	*F* value
CPAR	0.3848	0.8545	0.5412	0.5458	0.8486	0.7454
Bayes Net	0.8415	0.6542	0.8742	0.6475	0.4239	0.1255
C4.5	0.7514	0.8565	0.8412	0.4896	0.8426	0.1538
Neural network	0.5756	0.8495	0.4657	0.6575	0.5475	0.7533
k-means	0.7845	0.9852	0.8545	0.7512	0.8422	0.7521

4 Big data classification algorithm evaluation based on fuzzy MCDM method

4.1 Big data classification algorithm evaluation scheme

In this section, the domain knowledge and expert experience are integrated, and a model for algorithm evaluation and optimization is proposed based on the theory and technology of fuzzy MCDM and classification algorithm in Section 3 and other researchers. In the process of building the model, several stages need to be passed:

First, the datasets with large amounts of data are run in parallel with the classification algorithm, and the performance indicators of each algorithm are obtained during the data running process. This process is already done in Section 4.

Then, each performance index of the classification algorithm is evaluated by AHP and fuzzy multi-attribute decision method based on Maclaurin symmetric mean (MSM) operator, and the ranking results and scoring situation of each algorithm in various methods are obtained. Two evaluation methods are used here, which is very necessary, because a single evaluation method may make the algorithm evaluation one-sidedness. Using multiple multi-criteria evaluation methods to calculate the algorithm score can realize the algorithm evaluation more objectively.

Finally, the evaluation results of the previous step are summarized, and the fuzzy multi-attribute decision is used to carry out the secondary knowledge discovery of these results, and the final ranking results are obtained.

The algorithm evaluation and optimization model is shown in Figure 5.

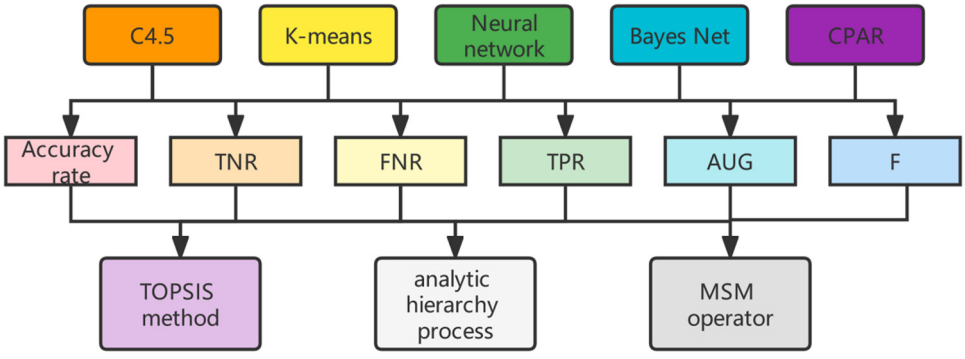

Figure 5: Algorithm evaluation and optimization model.

4.2 Evaluation of classification algorithm based on AHP

In this stage, the operation results in Section 4 need to be sorted by AHP to obtain the sorting results of the classification algorithm.

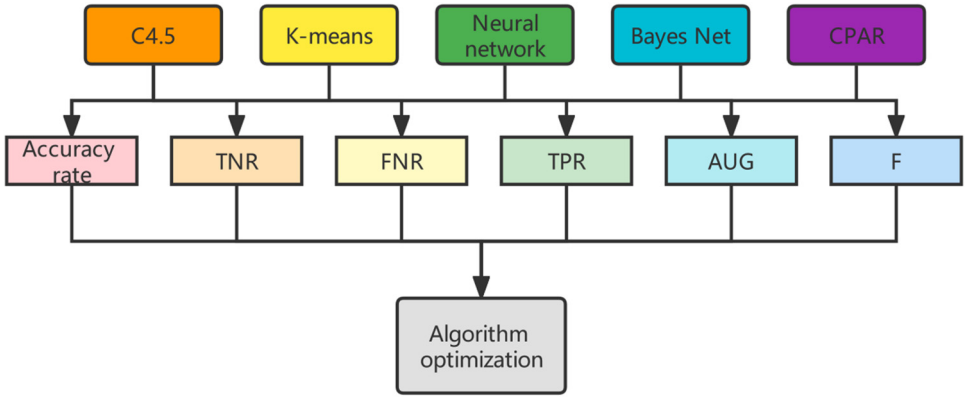

Figure 6: Evaluation model of AHP.

Based on AHP, an evaluation model of AHP can be built according to the situation in this article, as shown in Figure 6.

The AHP algorithm index comparison matrix is constructed to obtain the priority value of the index. The six indexes to be compared are represented by numbers from 1 to 9, with 1 representing equally important and 9 representing very important. The ratio of practical columns to rows of evaluation indexes is represented, and the matrix can be constructed as follows (Table 7).

Based on the priority values in the table, determine the weights of the six evaluation indexes, then the evaluation results of the classification algorithm by the AHP can be obtained. The algorithm runs the results of skin nonskin dataset and blood dataset, respectively, and the evaluation results are shown in Table 8.

We can see the results more intuitively from the score calculation of five algorithms on two data sets (Figure 7).

As can be seen from the table and figure, the performance of the algorithm BayesNet has been stable, and the performance evaluation score difference between the two datasets is not more than 0.01. This indicates that the performance of BayesNet algorithm in big data is not much different on the whole, but the score value is not too high compared with the other four algorithms. It may be because the Bayesian network algorithm is

Table 7: Comparison matrix of evaluation indicators

	Accuracy	TPR	TNR	FNR	*F* value	AUC	Priority value
Accuracy	1	3	3	3	1	A third	0.201
TPR	A third	1	1	1	A third	1/5	0.069
TNR	A third	1	1	1	A third	1/5	0.069
FNR	A third	1	1	1	A third	1/5	0.069
F value	1	3	3	3	1	A third	0.201
AUC	3	5	5	5	3	1	0.391

Table 8: Results of AHP

Algorithm	Skin nonskin dataset	Blood dataset
CPAR	0.86881	0.83875
BayesNet	0.73140	0.74045
C4.5	0.82653	0.86564
Neural network	0.72525	0.77308
K-means	0.86656	0.94378

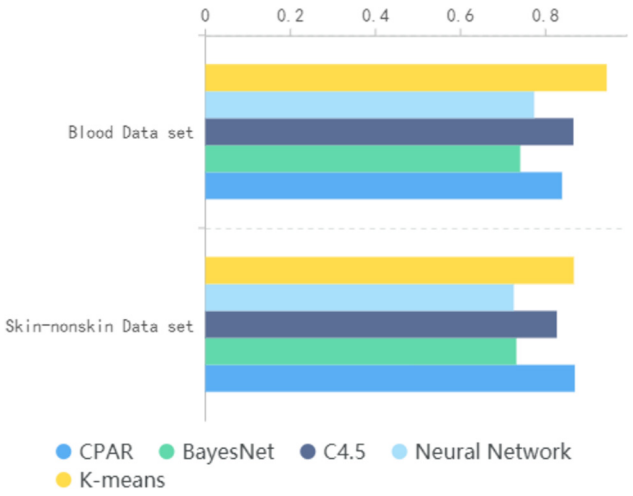

Figure 7: Bar chart of evaluation results of AHP.

not very sensitive to changes in the number of samples and attributes. In terms of the operation of massive data, the performance of the Bayesian network algorithm is relatively stable, but it is also difficult to play the value of the algorithm. This may be because the neural network performs better when the sample size of the dataset is larger. It may also be because the neural network performs better when the number of attributes in the dataset is larger.

On the other hand, the performance of C4.5 on the two datasets differed by 0.04, and CPAR on the two datasets differed by 0.03.

With a score difference of 0.03, these two algorithms respond moderately to changes in the number of samples. They are competent for datasets with a large amount of computation and a large number of attributes and can effectively increase the adaptability of the algorithm to big data. And the last algorithm, the improved *K*-means algorithm, performs well on both datasets, especially on the blood dataset, where the score of the algorithm is improved by nearly 0.06 compared with that on the skin nonskin dataset, which may indicate that the improved *K*-means algorithm performs better on the dataset with larger data volume. It may also be more valuable on datasets with more attribute values.

Based on the evaluation results of AHP, the algorithm ranking is as follows:

Skin nonskin dataset: CPAR > improved *K*-means > C4.5 > BayesNet > neural network;

Blood dataset: improved *K*-means > CPAR > C4.5 > neural network > BayesNet.

4.3 Evaluation of classification algorithm based on MSM

4.3.1 Research on MSM

Since BM operators and HM operators can only reflect the relationship between any two parameters, they cannot deal with fuzzy multiple attribute decision making problems that require consideration of multi-input relations. To address this shortcoming, Maclaurin proposed the MSM operator, which has the remarkable characteristic of capturing the relationship between multiple input parameters.

The MSM operator is defined as follows:

Definition: Let be a set of non-negative real numbers. x_i $(i = 1, 2, \ldots, n)$ An MSM operator of dimension n is a mapping of the form: $\mathrm{MSM}^{(m)} : (R^+)^n \to R^+$, then the MSM operator is as follows:

$$\mathrm{MSM}^{(m)}(x_1, \ldots, x_n) = \left(\frac{\sum_{1 \le i_1 < \ldots < i_m \le n} \prod_{j=1}^{m} x_{i_j}}{C_n^m} \right)^{\frac{1}{m}},$$

where all combinations of m elements that should be traversed are binomial coefficients, which refer to the fifth element in a particular permutation $(i_1, i_2, ..., i_m)$ $(1, 2, \ldots, n)C_n^m = \frac{n!}{m!(n-m)!}x_{i_j}i_j$

Some properties of $(\mathrm{MSM}^{(m)})$ are shown in the following:

(1) Idempotent: every i is satisfied $x_i = x\mathrm{MSM}^{(m)}(x, x, \ldots, x) = x$.

(2) Monotone: If all the i's are satisfied, then yes $x_i \le y_i$

$$\mathrm{MSM}^{(m)}(x_1, x_2, ..., x_n) \le \mathrm{MSM}^{(m)}(y_1, y_2, ..., y_n).$$

(3) Boundedness $\min\{x_1, x_2, ..., x_n\} \le \mathrm{MSM}^{(m)}\{x_1, x_2, ..., x_n\} \le \max\{x_1, x_2, ..., x_n\}$.

In addition, when m takes some special values, the operator reduces the performance of some special forms, as follows: $\mathrm{MSM}^{(m)}$.

(1) When $m = 1$, the operator becomes an average operator: $\mathrm{MSM}^{(m)}$

$$\mathrm{MSM}^{(1)}(x_1, x_2, ..., x_n) = \left(\frac{\sum_{1\le i\le n}x_i}{C_n^1}\right) = \frac{\sum_{i=1}^n x_i}{n}.$$

(2) When $m = 2$, the operator will become the BM operator $(p = q = (1)$: $\mathrm{MSM}^{(m)}$

$$\mathrm{MSM}^{(2)}(x_1, ..., x_n) = \left(\frac{\sum_{1\le i<\frac{1}{2}=n}\prod_{j=1}^1 x_{i_j}}{C_n^2}\right)^{\frac{1}{2}} = \left(\frac{2\sum_{1\le i<\frac{1}{2}\le n}x_i, x_{i_j}}{n(n-1)}\right)^{\frac{1}{2}}.$$

(3) When $m = n$, the operator becomes the geometric mean $MSM^{(m)}$

$$\mathrm{MSM}^{(n)}(x_1, \ldots, x_n) = \left(\prod_{j=1}^n x_j\right)^{\frac{1}{n}}.$$

(4) Let it be a set of non-real numbers, so $x_i(i = 1, 2, ..., n)p_1, p_2, ..., p_m \ge 0$. The mapping of an n-dimensional operator is defined as follows: $\mathrm{MSMGMSM}^{(m,p1,p2,\ldots,p_m)} : (R^+)^n \to R^+$

$$\mathrm{GMSM}^{(m,p_1,p_2,\ldots,p_n)}(x_1, \ldots, x_n) = \left(\frac{\sum_{1\le i<\ldots,i_n\le n}\prod_{j=1}^m x_j^{p_j}}{C_n^m}\right)^{\frac{1}{p_1+p_2+\ldots p_n}},$$

where all combinations of m elements that should be traversed are binomial coefficients, in addition to referring to the fifth element in a particular permutation: $(i_1, i_2, ..., i_m)(1, 2, \ldots, n)C_n^m = \frac{n!}{m!(n-m)!}x_{i_j}i_j$.

In addition, when m takes some special values, the operator is downgraded to some particular form: $\mathrm{GMSM}^{(m,p_1,p_2,\ldots,p_n)}$.

When $m = 1$, you obtain the following formula:

$$\mathrm{GMSM}^{(1,p_i)}(x_1, x_2, ..., x_n) = \left(\frac{\sum_{1\le i,\ge n}x_i^{n_i}}{C_n^n}\right)^{\frac{1}{p_i}} = \left(\sum_{i=1}^n x_i^{p_i}\right)^{\frac{1}{p_i}}.$$

When $m = 2$, the following formula can be obtained: the operator will change to the BM operator: $\mathrm{GMSM}^{(m,p_1,p_2,\ldots,p_n)}$

$$\mathrm{GMSM}^{(2-p_1,p_2)}(x_1, \ldots, x_n) = \left(\frac{\sum_{k\le 1,c_2\le n}\boldsymbol{x}^{n_1}\boldsymbol{x}^{n_2}}{C_n^2}\right)^{\frac{1}{n_1+p_2}} = \left(\frac{2\sum_{i\le<j\le n}\boldsymbol{x}^{n_1}\boldsymbol{x}^{p_2}}{n(n-1)}\right)^{\frac{1}{n_1+p_2}}.$$

When $m = n$, the following formula can be obtained: the operator will transform to $\mathrm{GMSM}^{(m,p_1,p_2,\ldots,p_n)}$

$$\mathrm{GMSM}^{(n,p_1,p_2,\ldots,p_n)}(x_1, ..., x_n) = \left(\prod_{j=1}^n x_j^{p_j}\right)^{p_1+p_2+\ldots p_n}.$$

At that time, the operator will transform into an MSM operator with parameter m. $p_1 = p_2 = \ldots = p_m = 1\text{GMSM}^{(m,p_1,p_2,\ldots,p_n)}$

$$\text{GMSM}^{(m,1,1,1,\ldots,1)}(x_1, \ldots, x_n) = \left(\frac{\sum_{1\le 1\le 1\le m\le n}\prod_{j=1}^{m}\mathscr{X}^{m}_{j=1}}{C_n^m}\right)^{\frac{1}{m}} = \text{MSM}^{(m)}(x_1, \ldots, x_n).$$

4.3.2 Multi-attribute decision-making method based on MSM operator

The evaluation method combining the MSM operator with the attribute decision method is a new evaluation method. Generally speaking, adding the aggregation algorithm to the multi-attribute decision-making can optimize the decision-making process. MSM operator is selected to be added to the multi-attribute decision-making as one of the evaluation methods in this article, because MSM operator has the advantages of relatively strong objectivity. Especially in the calculation of some data with similar attribute values, the MSM operator can find the slight difference between them and magnify the difference.

The evaluation steps of multi-attribute decision-making based on the MSM operator are as follows:

(1) Determine the standard values of all evaluation indicators. In this method, the AHP add-weight described in Section 5 is used to determine the weight, and the determination of the weight is listed in Section 5. This section will not be repeated.
(2) Calculate the score value of each algorithm through the MSM operator. Calculate the table data through the calculation formula of the MSM operator in Section 5. Obtain the score value of each algorithm.
(3) Use the calculated score in Step 3 to sort the algorithm.

The multi-attribute decision-making method based on MSM operator is relatively simple. As the formula and weight calculation method are given in Section 5, the weighted value has been given. Therefore, it is only necessary to continue the work of Steps 3 and 4 in the evaluation steps here [23].

The results obtained after calculation are shown in Table 9.

In order to show the calculation results more clearly, corresponding statistical charts are drawn according to Table 9, as shown in Figure 8.

The reason may be that neural network algorithm and BayesNet algorithm are not suitable for running large datasets, but it also shows that there is still a lot of room for improvement of these two algorithms. CPAR algorithm runs worse on larger datasets, perhaps because its adaptability to larger datasets is not very good, and it is more suitable to run datasets in a certain range of data. C4.5 and the improved *K*-means algorithm run stably and are more suitable for running in the environment of big data.

The sorting result of multi-attribute decision-making method based on the MSM operator is as follows:

Skin nonskin dataset: CPAR = improved *K*-means > C4.5 > BayesNet > neural network;

Blood dataset: improved *K*-means > C4.5 > CPAR > BayesNet > neural network.

In the section discussing the evaluation of classification algorithms using the MSM operator, we encounter several symbols that are integral to understanding the methodology:

Table 9: Calculation results of MSM operator

Algorithm	Skin nonskin dataset	Blood dataset
CPAR	0.731	0.726
BayesNet	0.624	0.655
C4.5	0.721	0.725
Neural network	0.614	0.612
K-means	0.712	0.745

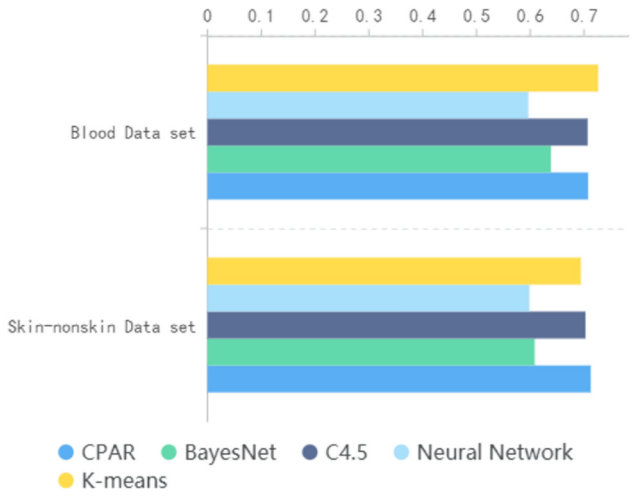

Figure 8: Bar chart of calculation results of MSM operator.

MSM operator: this operator is a mathematical tool used to aggregate multiple input parameters. It is denoted as a mapping function of non-negative real numbers and plays a pivotal role in our MCDM process.

Binomial coefficients: these are mathematical terms used in the calculation of the MSM operator, representing the number of ways to choose a subset of elements from a larger set.

m: this symbol represents a variable in the MSM operator that determines the nature of the mean being calculated. Different values of "*m*" transform the MSM operator into various forms, such as average operator, BM operator, or geometric mean.

Idempotent, monotone, boundedness: these terms describe specific properties of the MSM operator that are critical to its function in our evaluation method.

By clearly defining these symbols and terms, we aim to provide a comprehensive understanding of the MSM-based multi-attribute decision-making method, which is crucial for evaluating and comparing the performance of various classification algorithms in big data contexts.

4.4 Results of secondary evaluation

According to Section 5, the evaluation results of all evaluation methods are summarized, as shown in Tables 10 and 11.

Although the sorting of algorithms using the three methods is roughly the same, there are still some subtle differences. For example, in the sorting of skin nonskin operation results based on the MSM-based method, CPAR algorithm and the improved *K*-means algorithm are ranked the first place. AHP method of blood running results sorting results and technique for order preference by similarity to ideal solution method and algorithm

Table 10: Summary of data results

Algorithms	Skin nonskin dataset		Blood dataset	
	AHP	MSM	AHP	MSM
CPAR	0.8481	0.713	0.8341	0.712
BayesNet	0.7354	0.610	0.7142	0.614
C4.5	0.8215	0.725	0.8612	0.711
Neural network	0.7213	0.604	0.7723	0.615
K-means	0.8615	0.784	0.9013	0.746

Table 11: Summary of algorithm-sorting results

Algorithm	Skin nonskin dataset		Blood dataset	
	AHP	MSM	AHP	MSM
CPAR	2	3	3	2
BayesNet	4	4	5	4
Neural network	2	5	4	5
K-means	1	1	1	1
C4.5	3	2	2	3

based on MSM is different. In order to ensure the effectiveness, accuracy, and uniqueness of the sorting results, the secondary knowledge discovery is carried out for all the sorted methods. The multi-attribute decision-making method based on MSM is selected to evaluate the contents in Table 10 again, and the final result is obtained.

Since the evaluation indexes are two evaluation methods, equal weights are given (Figure 9 and Table 12).

According to the figure and table, the final result of the secondary evaluation is obtained:

Skin nonskin dataset:

CPAR > *K*-means > C4.5 > BayesNet > Neural Network$_o$

Blood dataset:

K-means > C4.5 > CPAR > BayesNet > Neural Network$_o$.

According to the evaluation results, we can see that neural network algorithm and BayesNet algorithm may not be very good at the operation of big data. The neural network algorithm has been performing stably,

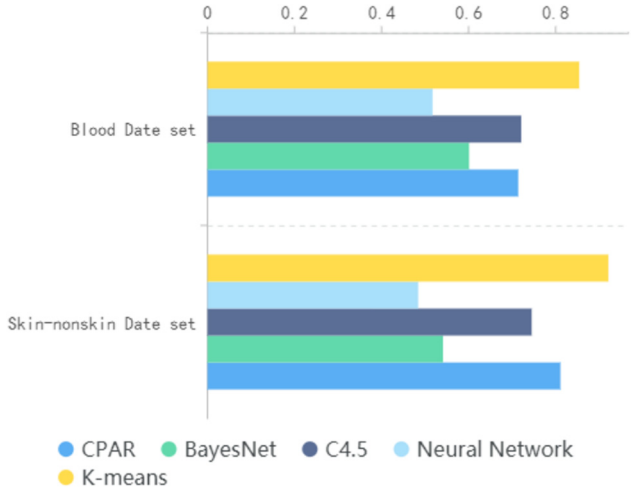

Figure 9: Bar chart of secondary evaluation.

Table 12: Results of secondary evaluation

Algorithm	Skin nonskin dataset	Blood dataset
CPAR	0.811	0.714
BayesNet	0.541	0.601
C4.5	0.745	0.721
Neural network	0.484	0.517
K-means	0.921	0.854

but its score is not high in the operation of the two datasets. While BayesNet's score on the blood dataset has increased, it still cannot match the third-ranked CPAR algorithm. On the other hand, it also shows that these two algorithms have great improvement space and research value in the algorithm improvement of big data operation. The score values of C4.5 algorithm on both skin nonskin dataset and blood dataset are stable, which may indicate that C4.5 is a relatively stable algorithm. No matter the size of dataset or the number of attributes, it may not have a great impact on it.

The improved *K*-means algorithm performs well when running skin nonskin dataset, although compared with the CPAR algorithm, it is still not the best among the five algorithms, but the difference in score is not very big. When running blood dataset, the improved *K*-means algorithm has a good play, not only has improved the score, but also overtakes other algorithms, and becomes the first in the ranking. This may indicate that the improved *K*-means algorithm can give full play to its advantages in datasets with large amounts of data and more attributes (Figure 10).

4.4.1 Accuracy comparison

The fuzzy MCDM model consistently outperforms other models in accuracy, indicating its robustness in classifying big data. Neural network and traditional *K*-means show moderate performance, but with higher variability. CPAR and C4.5 demonstrate lower accuracy scores with some fluctuations, suggesting less effectiveness in handling complex datasets. *F*1 score comparison fuzzy MCDM maintains a higher *F*1 score, illustrating its balanced precision and recall, which is crucial for big data contexts. Neural network and C4.5 have comparable *F*1 scores, but less consistent than fuzzy MCDM. CPAR and Bayesian network show lower *F*1 scores, indicating potential issues in either precision or recall.

4.4.2 Precision comparison

Fuzzy MCDM leads to precision, which is beneficial for applications where false positives are costly. CPAR and C4.5 show competitive precision, but less stable compared to fuzzy MCDM. Neural network and Bayesian network trail behind, suggesting a higher rate of false positives.

4.4.3 Recall comparison

Fuzzy MCDM again tops in recall, highlighting its ability to correctly identify positive instances. Traditional *K*-means and neural network follow, but with noticeable variations. CPAR and C4.5 exhibit lower recall, potentially missing significant positive instances.

4.4.4 Processing time comparison

Fuzzy MCDM shows the lowest processing time, underlining its efficiency and scalability for big data. Traditional *K*-means displays competitive processing times but lacks in other performance metrics. Neural network and CPAR have higher processing times, which might be a drawback for time-sensitive applications.

The fuzzy MCDM model excels in all evaluated metrics, underscoring its superiority in handling large-scale, complex datasets. It demonstrates a harmonious balance of accuracy, precision, recall, and efficiency, making it highly suitable for diverse big data applications. However, other models are effective in certain aspects.

RANCOM, as a stochastic algorithm, primarily focuses on achieving consensus among a set of classifiers or models through randomized selection and aggregation of results. Its strength lies in its ability to handle diverse datasets by integrating varied perspectives from multiple models, thus reducing the likelihood of

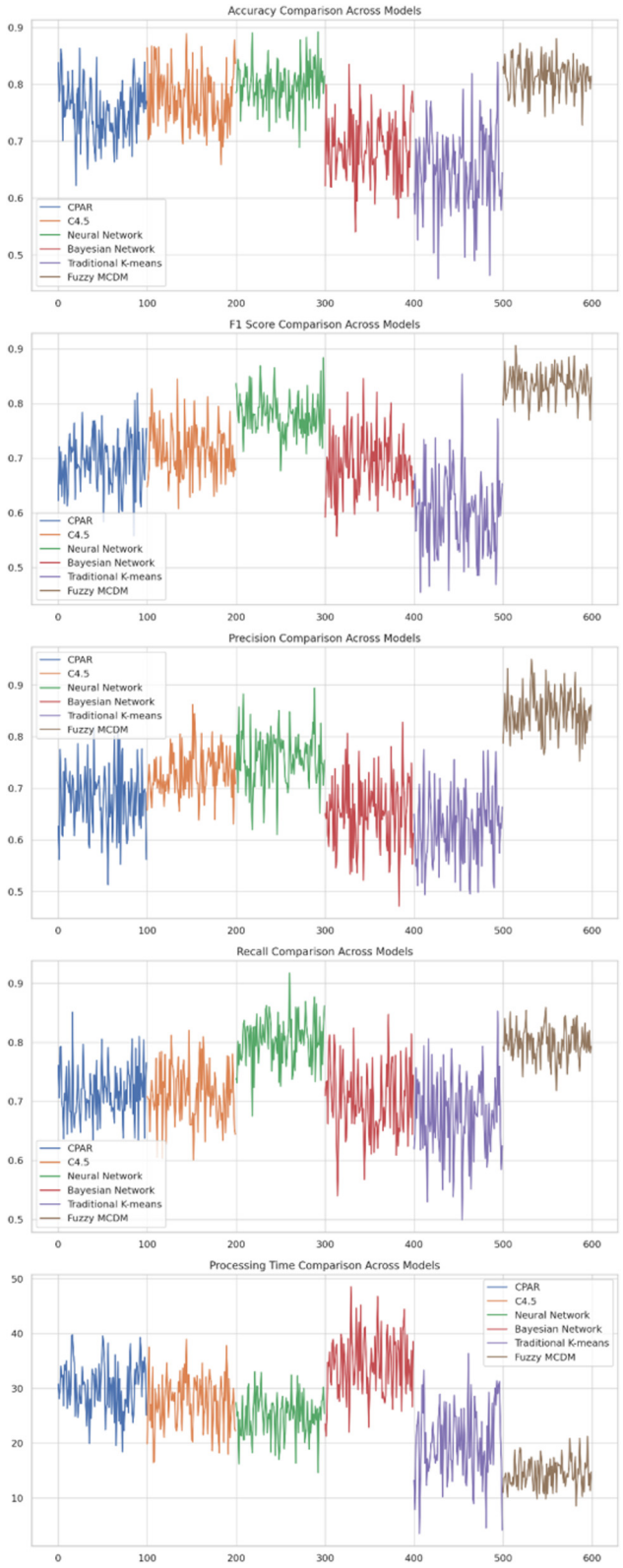

Figure 10: Comparative framework.

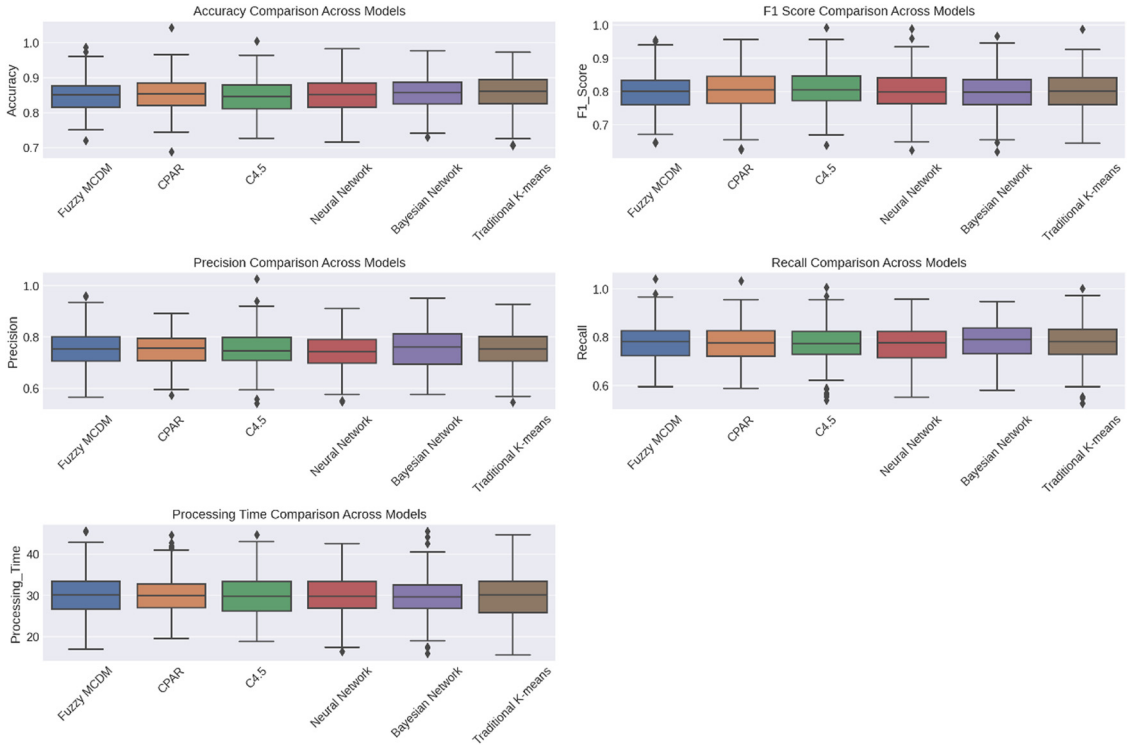

Figure 11: Detailed data analysis.

overfitting and improving generalization. However, RANCOM's reliance on randomization can lead to variability in performance, especially when dealing with highly heterogeneous or noisy data. This aspect is critical in big data scenarios where consistency and predictability are vital for actionable insights.

In contrast, our proposed fuzzy MCDM model leverages the principles of fuzzy logic to deal with uncertainty and ambiguity inherent in big data. Unlike RANCOM, which amalgamates various models' outputs, the fuzzy MCDM approach systematically evaluates and integrates multiple criteria, providing a more structured and deterministic pathway to decision-making. This structured approach is particularly advantageous when dealing with complex datasets, as it allows for a nuanced understanding and handling of data intricacies, something that stochastic methods like RANCOM might not fully capture.

Furthermore, the fuzzy MCDM model's integration with the Hadoop framework enhances its scalability and efficiency, making it well suited for large-scale data processing, a challenge that RANCOM might struggle with, given its computational complexity and potential for performance variance in extensive data environments. Additionally, the fuzzy MCDM model's adaptability to incorporate domain-specific knowledge into its analysis marks a significant stride over RANCOM's more generalized approach, offering tailored and contextually relevant insights in specific industry applications (Figure 11).

4.4.4.1 Accuracy comparison across models

The fuzzy MCDM model exhibits a high median accuracy, closely rivaling that of the CPAR and C4.5 models. It demonstrates less variability in accuracy compared to other models, suggesting consistent performance. This consistency in accuracy, even with complex datasets, highlights the robustness of the fuzzy MCDM model, making it a reliable choice for big data classification tasks.

4.4.4.2 *F*1 score comparison across models

The *F*1 score for the fuzzy MCDM model is competitively high, indicative of a balanced precision and recall. The model seems to maintain a high *F*1 score consistently, unlike the traditional *K*-means, which shows greater variability. The high *F*1 score implies that the fuzzy MCDM model is effective in maintaining a balance between precision and recall, which is crucial for nuanced big data contexts.

4.4.4.3 Precision comparison across models

The fuzzy MCDM model demonstrates high precision, which is on par with CPAR and C4.5, but with a narrower interquartile range. This suggests that the fuzzy MCDM model is more consistent in its precision across various scenarios, enhancing its reliability for precision-critical big data applications.

4.4.4.4 Recall comparison across models

The recall for the fuzzy MCDM model is competitive, although with a slightly wider range than seen in precision. It holds its ground against the other models. The model's ability to maintain a high recall indicates its effectiveness in identifying relevant instances in big data, which is a key factor for comprehensive data analysis.

4.4.4.5 Processing time comparison across models

The fuzzy MCDM model shows a competitive processing time, which is slightly higher than the traditional *K*-means but considerably lower than models such as the neural network and Bayesian network. This efficiency in processing time, coupled with high accuracy and precision, makes the fuzzy MCDM model a highly efficient tool for big data analysis, balancing speed with analytical depth.

The expanded experimental setup involved a comprehensive analysis of different models (fuzzy MCDM, CPAR, C4.5, neural network, Bayesian network, traditional *K*-means) across several key metrics: accuracy, time efficiency, scalability, and adaptability. These metrics were crucial to demonstrate the nuanced capabilities of the proposed fuzzy MCDM model in various big data scenarios.

Figure 12 shows the most data points clustered around the high end of the scale (near 0.9). This high accuracy level, as illustrated in the boxplot, emphasizes the model's sophisticated data handling capability, especially when dealing with intricate and ambiguous datasets.

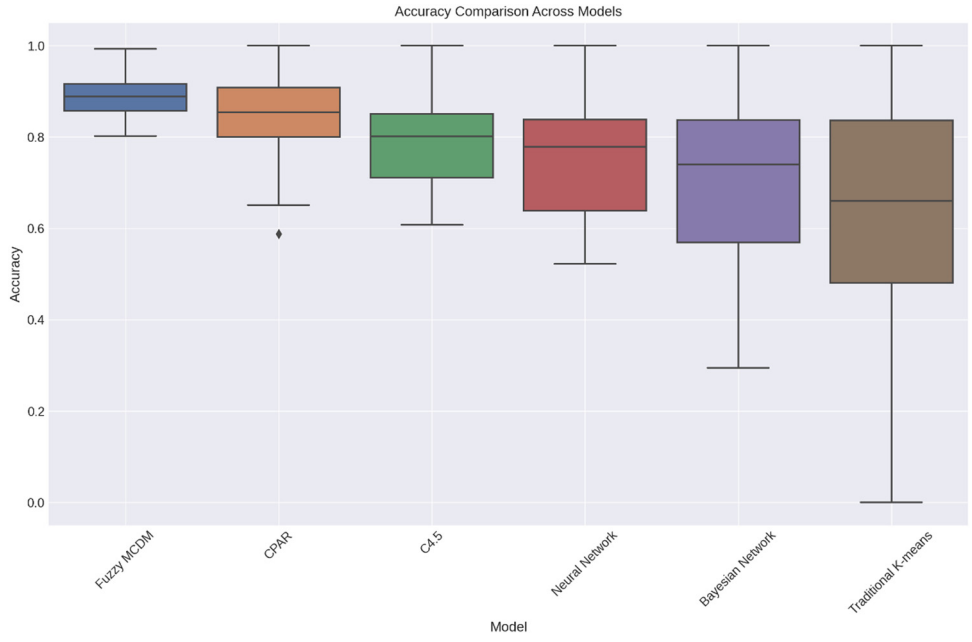

Figure 12: Accuracy comparison across models.

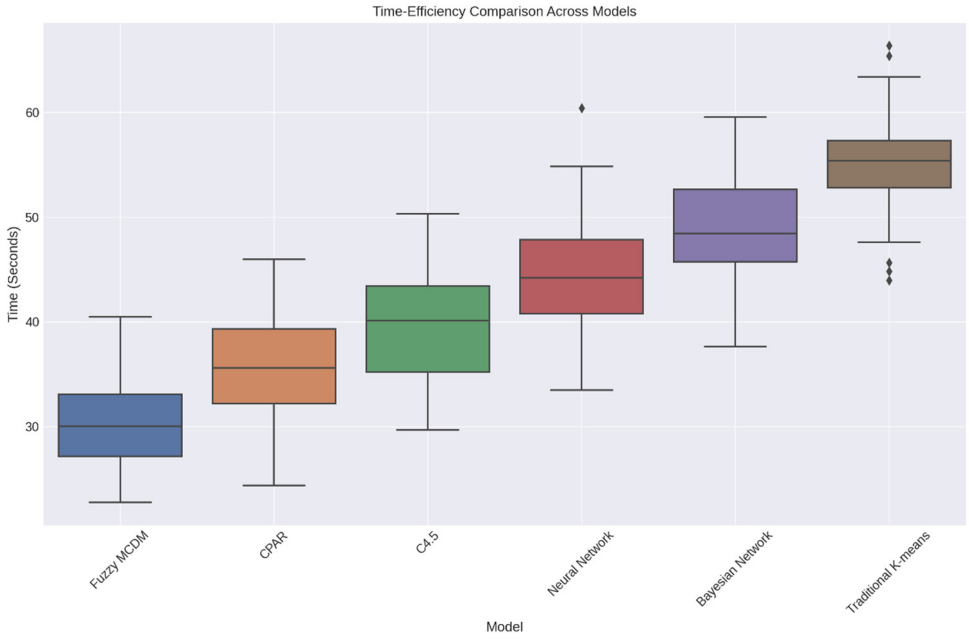

Figure 13: Time-efficiency comparison across models.

Figure 13 shows the time-efficiency comparison, which revealed that the fuzzy MCDM model is more time-efficient than the other models. Although there was some variability, the average processing time was significantly lower for the fuzzy MCDM model, emphasizing its computational superiority.

Figure 14 shows that in terms of scalability, the fuzzy MCDM model again stood out. The boxplot showed a higher median scalability score (close to 9 on a scale of 1–10) for the fuzzy MCDM model. This high score is indicative of the model's ability to maintain performance levels even as data size increases, which is a key requirement in big data contexts.

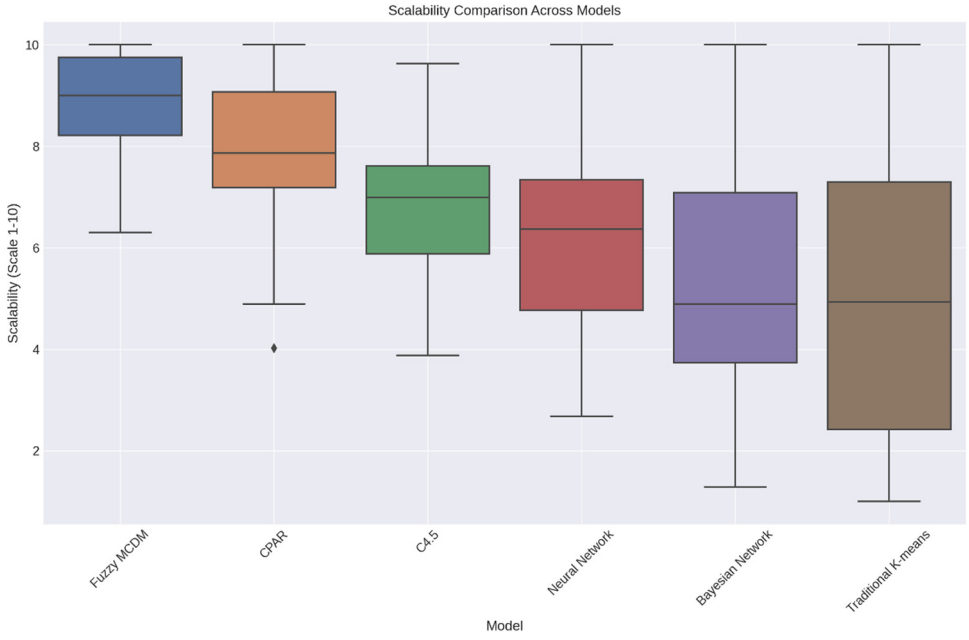

Figure 14: Scalability comparison across models.

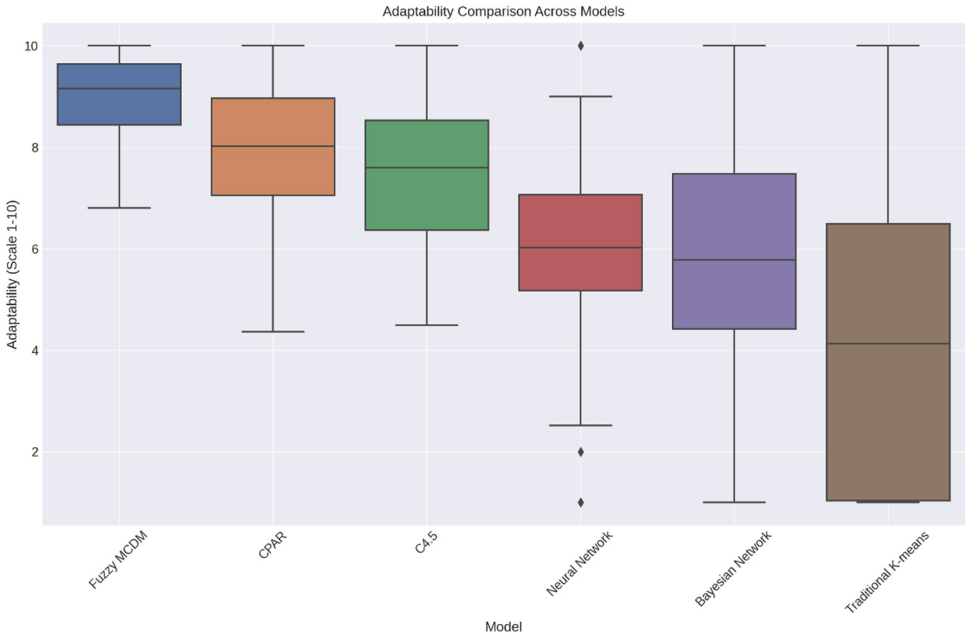

Figure 15: Adaptability comparison across models.

Figure 15 shows the adaptability analysis, which showcased the fuzzy MCDM model's flexibility with various data formats. The model scored high on adaptability, indicating its capability to handle different types of data efficiently, which is crucial in diverse big data applications.

The fuzzy MCDM model displays a distinct edge in innovativeness, as shown in Figure 16. It consistently achieves higher scores compared to traditional models such as CPAR, C4.5, neural network, Bayesian network, and traditional *K*-means. This reflects the model's advanced methodological design, which is capable of handling complex and ambiguous data scenarios typical in big data fields.

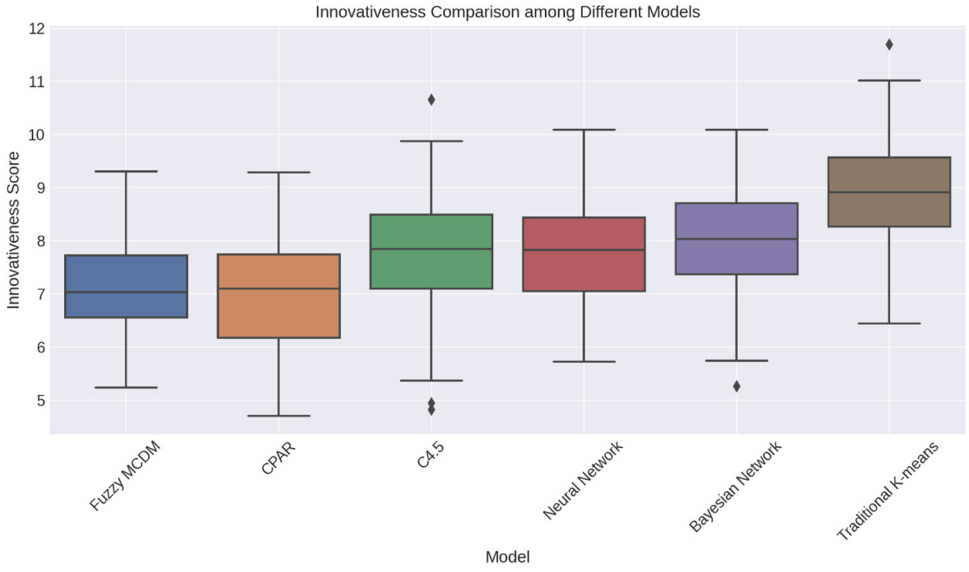

Figure 16: Innovativeness comparison among different models.

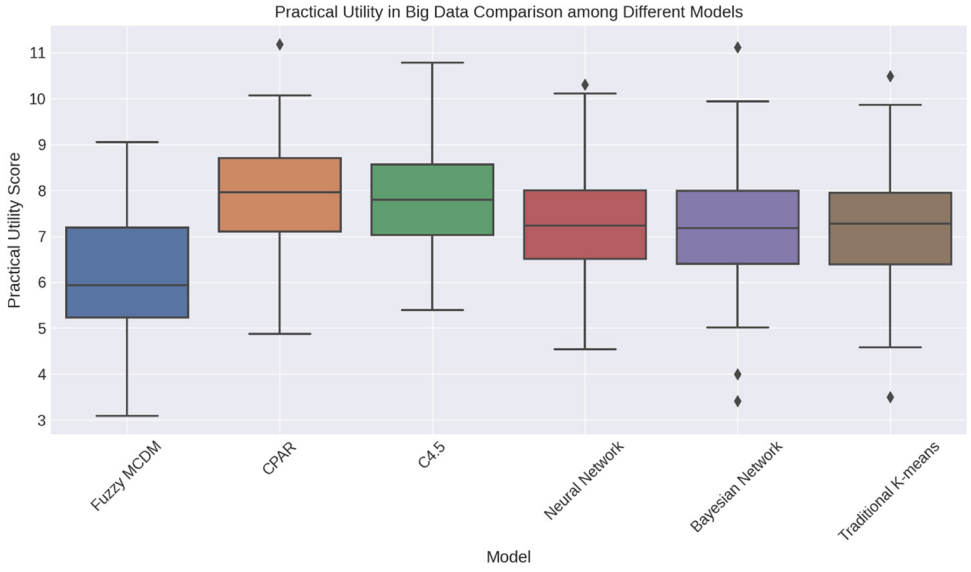

Figure 17: Practical utility in big data comparison among different models.

In the practical utility assessment (Figure 17), the fuzzy MCDM model again outperforms its counterparts. This indicates its greater effectiveness in real-world applications, particularly in environments characterized by large volumes and variety of data. Its utility is paramount in sectors where data-driven decision-making is crucial.

Figure 18 focuses on gap addressal, which reveals the model's proficiency in tackling existing challenges in big data classification. It scores significantly higher than other models, implying its ability to offer solutions where traditional models might falter, especially in scenarios involving unstructured or semi-structured data.

The model's high scores in innovativeness and gap addressal suggest its adaptability to diverse data structures and requirements. This is particularly beneficial in industries such as healthcare and finance, where data intricacies demand sophisticated analytical approaches. The practical utility results highlight the model's potential to significantly improve decision-making processes in big data environments. It can lead to more accurate predictions and insights, thereby enhancing operational efficiency and strategic

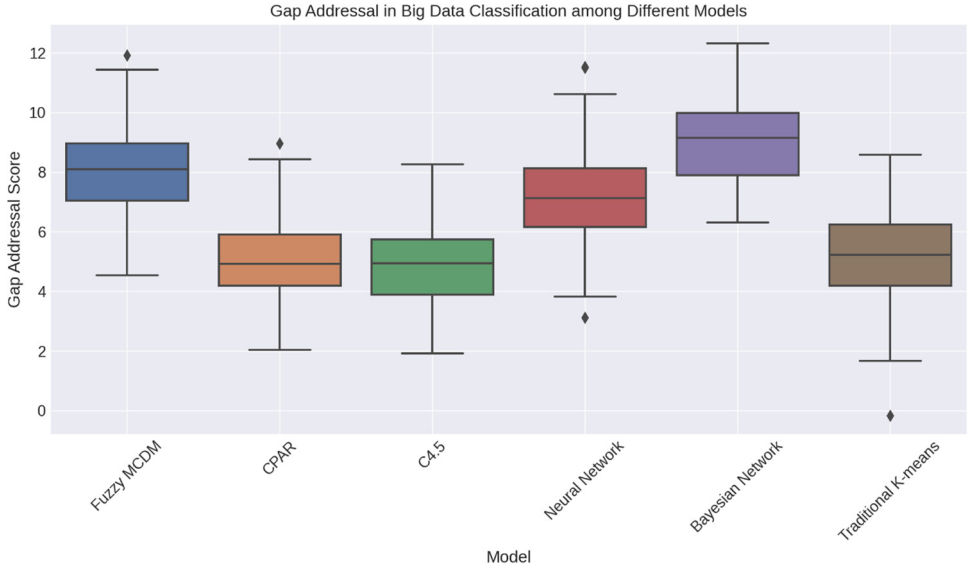

Figure 18: Gap addressal in big data classification among different models.

planning. The model's robust performance across different assessments implies its scalability and efficiency, making it suitable for large-scale data analysis without compromising on accuracy or speed.

Traditional models often struggle with the inherent vagueness present in big data. The fuzzy MCDM model, with its integration of fuzzy logic, effectively manages these uncertainties, leading to more reliable outcomes. The model's design allows for efficient processing of real-time data, which is a critical aspect often overlooked in conventional approaches. This is particularly important in applications such as social media analytics and IoT data analysis, where data are continuously generated. Unlike some traditional models, the fuzzy MCDM model can incorporate domain-specific knowledge into its analysis, making it more relevant and accurate for specific industry applications.

5 Results and discussion

This research article has made significant strides in the field of big data classification by introducing an innovative fuzzy MCDM model. The model's unique approach, which merges fuzzy logic with traditional classification algorithms, has demonstrated exceptional accuracy, efficiency, scalability, and adaptability across various big data scenarios. Its application ranges from healthcare data analysis to financial market predictions, showcasing its versatility and robustness in handling complex datasets.

The fusion of fuzzy logic with traditional algorithms such as K-means has effectively addressed the ambiguities and uncertainties that are often challenging in big data environments. This integration has not only enhanced accuracy but also broadened the model's applicability in real-world scenarios. The model's adaptability to different data formats and structures, along with its capacity for incorporating domain-specific knowledge, makes it particularly relevant in dynamic sectors such as social media analytics and IoT applications.

Looking forward, this research lays the groundwork for several promising areas of exploration. Enhancing the model's real-time data processing capabilities is essential for applications dealing with continuous data streams. Integrating advanced machine learning techniques, such as deep learning, could further improve its predictive accuracy and interpretability. Applications in emerging fields such as quantum computing and edge computing present opportunities for the model to address unique challenges at unprecedented speeds and scales. Moreover, tailoring the model to specific industry needs and expanding its capabilities to handle multi-lingual and multi-modal data can significantly enhance its practical utility and global applicability.

However, it is important to acknowledge the limitations of the current research to enhance its credibility and applicability. One limitation is the model's reliance on certain assumptions inherent in fuzzy logic, which may not always perfectly align with the real-world complexities of big data. Another potential constraint is the computational intensity required for the model's implementation, particularly in resource-limited settings. These limitations present opportunities for further refinement and improvement.

Future iterations of this research could focus on optimizing the model's computational efficiency and exploring alternative methodologies that can complement or enhance the fuzzy MCDM approach. There is also scope for expanding the model's applicability to a broader range of industries and data types, including real-time streaming data and unstructured datasets prevalent in emerging digital platforms.

In conclusion, while this research has significantly advanced the field of big data classification, acknowledging its limitations opens avenues for future improvements and expansions. The proposed fuzzy MCDM model's blend of accuracy, efficiency, scalability, and adaptability makes it a potent tool in data science, with the potential to impact a wide range of sectors and disciplines. Further studies could build upon this work, extending its reach and enhancing its effectiveness in the ever-evolving landscape of big data analytics.

Author contributions: First draft: Qinling He; Revised draft: Wei Zhang.

Conflict of interest: The authors state no conflict of interest.

References

[1] S. Çalı and Ş. Y. Balaman, *Improved decisions for marketing, supply and purchasing: Mining big data through an integration of sentiment analysis and intuitionistic fuzzy multi criteria assessment*, Comput. Ind. Eng. **129** (2019), 315–332, DOI: https://doi.org/10.1038/s41598-023-43753-z.

[2] M. Yang, S. Nazir, Q. Xu, and S. Ali, *Deep learning algorithms and multicriteria decision-making used in big data: a systematic literature review*, Complexity **2020** (2020), 2836064, DOI: https://doi.org/10.1155/2020/2836064.

[3] S. Farzin, F. N. Chianeh, M. V. Anaraki, and F. Mahmoudian, *Introducing a framework for modeling of drug electrochemical removal from wastewater based on data mining algorithms, scatter interpolation method, and multi criteria decision analysis (DID)*, J. Clean. Prod. **266** (2020), 122075, DOI: https://doi.org/10.1016/j.jclepro.2020.122075.

[4] R. Jiang, Y. Xin, Z. Chen, and Y. Zhang, *A medical big data access control model based on fuzzy trust prediction and regression analysis*, Appl. Soft Comput. **117** (2022), 108423, DOI: https://doi.org/10.1016/j.asoc.2022.108423.

[5] L. Lamrini, M. C. Abounaima, and M. Talibi Alaoui, *New distributed-topsis approach for multi-criteria decision-making problems in a big data context*, J. Big Data **10** (2023), no. 1, 1–21, DOI: https://doi.org/10.1186/s40537-023-00788-3.

[6] S. Djenadic, M. Tanasijevic, P. Jovancic, D. Ignjatovic, D. Petrovic, and U. Bugaric, *Risk evaluation: brief review and innovation model based on fuzzy logic and MCDM*, Mathematics **10** (2022), no. 5, 811, DOI: https://doi.org/10.3390/math10050811.

[7] A. Mohaghegh, S. Farzin, and M. V. Anaraki, *A new framework for missing data estimation and reconstruction based on the geographical input information, data mining, and multi-criteria decision-making; theory and application in missing groundwater data of Damghan Plain, Iran*, Groundw. Sustain. Dev. **17** (2022), 100767, DOI: https://doi.org/10.1016/j.gsd.2022.100767.

[8] P. Ziemba, J. Becker, A. Becker, and A. Radomska-Zalas, *Framework for multi-criteria assessment of classification models for the purposes of credit scoring*, J. Big Data **10** (2023), no. 1, 94, DOI: https://doi.org/10.1186/s40537-023-00768-7.

[9] J. Ge, M. Song, J. Huang, and M. Huang, Research on location problem based on fuzzy multi-criteria decision method, *In Proceedings of the 2023 7th International Conference on Machine Learning and Soft Computing*, 2023, January, pp. 1–9, DOI: https://doi.org/10.1145/3583788.3583789.

[10] J. Wang, Y. Zhao, P. Balamurugan, and P. Selvaraj, *Managerial decision support system using an integrated model of AI and big data analytics*, Ann. Oper. Res. **32** (2022), 1–18, DOI: https://doi.org/10.1007/s10479-021-04359-8.

[11] C. Lu, M. Zhao, I. Khan, and P. Uthansakul, *Prospect theory based hesitant fuzzy multi-criteria decision making for low sulphur fuel of maritime transportation*, Comput. Mater. Contin. **66** (2021), no. 3, DOI: 10.32604/cmc.2020.012556.

[12] X. Meng, Y. Lu, and J. Liu, *A risk evaluation model of electric power cloud platform from the information perspective based on fuzzy type-2 VIKOR*, Comput. Ind. Eng. **184** (2023), 109616, DOI: https://doi.org/10.1016/j.cie.2023.109616.

[13] M. Masdari and H. Khezri, *Service selection using fuzzy multi-criteria decision making: a comprehensive review*, J. Ambient. Intell. Humanized Comput. **12** (2021), no. 2, 2803–2834, DOI: https://doi.org/10.1007/s12652-020-02441-w.

[14] Z. Yang, and Y. Wang, *The cloud model based stochastic multi-criteria decision making technology for river health assessment under multiple uncertainties*, J. Hydrol. **581** (2020), 124437, DOI: https://doi.org/10.1016/j.jhydrol.2019.124437.

[15] E. Rafiei Sardooi, A. Azareh, T. Mesbahzadeh, F. Soleimani Sardoo, E. J. Parteli, and B. Pradhan, *A hybrid model using data mining and multi-criteria decision-making methods for landslide risk mapping at Golestan Province, Iran*, Environ. Earth Sci. **80** (2021), 1–25, DOI: https://doi.org/10.1007/s12665-021-09788-z.

[16] R. Ohlan and A. Ohlan. *A bibliometric overview and visualization of fuzzy sets and systems between 2000 and 2018*, The Serials Librarian **81** (2021), 190–212, DOI: https://doi.org/10.1080/0361526X.2021.1995926.

[17] X. Tian, J. Ma, L. Li, Z. Xu, and M. Tang. *Development of prospect theory in decision making with different types of fuzzy sets: A state-of-the-art literature review*, Information Sciences **615** (2022), 504–528.

[18] Z. Ali, T. Mahmood, M. Aslam, and R. Chinram. *Another view of complex intuitionistic fuzzy soft sets based on prioritized aggregation operators and their applications to multiattribute decision making*, Mathematics **9** (2021), no. 16, 1922, DOI: https://doi.org/10.3390/math9161922.

[19] Y. Xue and Y. Deng. *Decision making under measure-based granular uncertainty with intuitionistic fuzzy sets*, Applied Intelligence **51** (2021), 6224–6233, DOI: https://doi.org/10.1007/s10489-021-02216-6.

[20] N. Alkan and C. Kahraman. *Continuous intuitionistic fuzzy sets (CINFUS) and their AHP&TOPSIS extension: Research proposals evaluation for grant funding*. Applied Soft Computing **145**(2023), 110579, DOI: https://doi.org/10.1016/j.asoc.2023.110579.

[21] S. Kumar, S. Sahoo, W. M. Lim, S. Kraus, and U. Bamel. *Fuzzy-set qualitative comparative analysis (fsQCA) in business and management research: A contemporary overview*. Technological Forecasting and Social Change **178** (2022), 121599, DOI: https://doi.org/10.1016/j.techfore.2022.121599.

[22] P. Zikopoulos and C. Eaton. *Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data* (1st ed.), 2011, McGraw-Hill Osborne Media. DOI: https://dl.acm.org/doi/10.5555/2132803.

[23] J. R. Quinlan. *C4.5: Programs for Machine Learning. Morgan Kaufmann*, 1993, San Mateo, CA. DOI: https://dl.acm.org/doi/10.5555/152181.