Migration Letters

Volume: 21, No: 5, pp. 751-768

ISSN: 1741-8984 (Print) ISSN: 1741-8992 (Online)
www.migrationletters.com

Knowledge Fusion by Harnessing Support Vector Machines for Collaborative Uncertain Data Classification in Multiagent Systems

Ahmad MohdAziz Hussein[1], Rashed M H Al-azzeh[2], Ala Mughaid[3], Raed Abu Zitar[4], Hazem Migdady[5], Laith Abualigah[6]

Abstract

Distributed data mining (DDM) has emerged as a useful method for analyzing data that is spread across multiple sources. Nevertheless, DDM has other challenges that restrict its effectiveness, such as autonomy, privacy, efficiency, and implementation. DDM's rigidity and lack of adaptability may render it unsuitable for numerous applications due to its requirement for a consistent environment, administration, control, and categorization procedures. In order to address these challenges, we suggest the implementation of MAS-DDM, which combines a multiagent system (MAS) with DDM. MAS, or Multiagent Systems, is a methodology used to create independent agents that possess shared environments and can collaborate and communicate with one another. The study showcases the advantages and attractiveness of MAS-DDM. In the context of MAS-DDM, agents can exchange their thoughts, even when the data they possess is classified and cannot be disclosed. Other agents can then decide whether to incorporate these beliefs into their decision-making process, which may result in a revision of their initial assumptions about each data class. MAS-DDM focuses on the support vector machine (SVM) method, which is commonly employed for handling uncertain data. Our investigation demonstrates that the performance of MAS-DDM surpasses that of DDM strategies that do not incorporate communicative processes, even when all MAS-DDM agents utilize the same methodology. We present empirical evidence demonstrating that the precision of the categorization job is significantly enhanced through the exchange of knowledge among agents.

Keywords: *Distributed Data Mining, Multiagent Systems, MAS-DDM, Support Vector Machines, Uncertain Data, Knowledge Sharing.*

1. Introduction

In the age of extensive and diversified data sources, distributed data mining (DDM) has become a crucial technique for extracting valuable insights from information scattered across several platforms [1,2]. Although DDM offers a persuasive methodology, it has obstacles that hinder its efficacy, such as concerns over independence, confidentiality, productivity, and execution. The inflexible characteristics of DDM, which need a

[1] Department of Computer Science, Faculty of Information Technology, Middle East University, Amman, Jordan
[2] Department of Computer Science, Faculty of Information Technology, Middle East University, Amman, Jordan
[3] Department of Information Technology, Faculty of Prince AL-Hussein Bin Abdullah II of Information Technology, The Hashemite University, P.O box 330127, Zarqa, 13133, Jordan
[4] Sorbonne Center of Artificial Intelligence, Sorbonne University-Abu Dhabi, Abu Dhabi, UAE
[5] CSMIS Department, Oman College of Management and Technology, 320 Barka, Oman
[6] Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan.

standardized environment and administrative supervision, restrict its usefulness in different situations, prompting the need to explore creative solutions [3,4].

With the abundance of numerous data sources in today's world, distributed data mining (DDM) has become a crucial way for extracting valuable insights from information scattered across several platforms [5,6]. Although DDM offers a persuasive strategy, it has obstacles that hinder its efficacy, such as concerns over independence, confidentiality, productivity, and execution. The inflexible characteristics of DDM, which need a uniform setting and administrative authority, restrict its usefulness in different situations, prompting the need to investigate creative alternatives [7].

With the integration of MAS-DDM (Multiagent System integrated with Distributed Data Mining), it is crucial to prioritize addressing security and privacy concerns in the evolving landscape of collaborative knowledge discovery [8]. The innovative methodology of MAS-DDM introduces significant revolutionary powers, yet its collaborative nature gives rise to important concerns that require careful and thorough consideration. Ensuring security and privacy are of utmost importance in the current data analysis environment, especially when handling sensitive information from many sources. The collaborative structure of MAS-DDM requires strong measures to protect against unwanted access and data breaches while also promoting knowledge sharing across agents [9].

Within the field of artificial intelligence and distributed computing, multiagent systems (MAS) are distinguished as a model that embodies the fundamental concept of collaborative autonomy. Multiagent systems aim to replicate the actions of independent entities, referred to as agents, that function inside a common environment, demonstrating the ability to collaborate, communicate, and make flexible decisions. A multiagent system is fundamentally composed of a group of agents, each with distinct objectives, knowledge, and abilities. Contrary to conventional systems with only one agent, a multiagent system (MAS) takes advantage of the combined effect produced by the interactions among multiple independent agents. This allows them to cooperate and jointly pursue common goals or individual aims within a shared framework. The concept of multiagent systems has become prominent because of its wide applicability in various fields, including artificial intelligence, robotics, economics, and social sciences. MAS provides a versatile and adaptable method for solving problems in which the combined intellect of agents surpasses the capacity of individual entities. The collaborative aspect of this corresponds perfectly with the complexities of real-world situations where varied entities need to interact and adjust dynamically [10].

In our study environment, autonomous agents, which are at the forefront of artificial intelligence and robotics, exemplify the fundamental concepts of intelligent autonomy. These entities, regardless of whether they are actual or virtual, are distinguished by their capacity to function independently, showcasing self-regulation, flexibility, and instantaneous decision-making. In my research, autonomous agents are crucial in the collaborative dynamics of Multiagent Systems (MAS). They enhance the system's collective intelligence by navigating their environment, processing information, and making adaptive decisions. The incorporation of self-governing entities into the Multiagent System (MAS) framework reveals a revolutionary capability, allowing the system to address intricate issues in ever-changing and uncertain surroundings where conventional methods may be inadequate. The unique characteristics of autonomous agents, including perception, decision-making, adaptability, interactivity, and learning, play a crucial role in building the intelligent structure of MAS. My research is to explore the complex interactions between these independent entities in order to discover novel approaches to problem-solving, decision-making, and collaborative intelligence in a shared environment. The study of self-governing entities within the Multiagent System (MAS) framework is essential for comprehending the cooperative independence that drives the system's abilities. This makes it an intriguing approach for tackling various challenges in different fields, such as robotics and information systems.

The advent of distributed data mining (DDM) has become a valuable method in modern data analysis for studying data that is spread across various sources. Nevertheless, the effectiveness of DDM is limited by a range of obstacles that involve concerns of independence, confidentiality, efficiency, and execution. The inherent rigidity and inflexibility of DDM can make it impractical for different uses, as it requires a consistent setting, administrative structure, control mechanisms, and classification procedures.

In order to address these difficulties and improve the flexibility of data mining techniques, we suggest a new method called MAS-DDM, which involves the seamless integration of a Multiagent System (MAS) with distributed data mining. Multiagent Systems (MAS) are a methodology used to develop autonomous agents that have shared environments. MAS enables agents to cooperate and communicate with each other. The objective of our work is to demonstrate the significant benefits and attractiveness of MAS-DDM as a novel approach to address the constraints of conventional DDM.

In the context of MAS-DDM, the exchange of information between agents is of utmost importance, especially in situations when the data is sensitive and cannot be publicly disclosed. Agents in MAS-DDM can express and share their opinions, enabling other agents to make well-informed decisions on whether to integrate these beliefs into their decision-making processes. The collaborative methodology facilitates a dynamic reassessment of initial assumptions pertaining to each data class, thereby overcoming the challenges of autonomy and privacy that frequently impede traditional DDM approaches.

The core of MAS-DDM revolves around the application of the support vector machine (SVM) technology, which is a commonly used approach for dealing with uncertain data. Our analysis confirms that MAS-DDM consistently achieves better results compared to DDM techniques that do not have communicative mechanisms, even when all MAS-DDM agents use the same SVM algorithm. The empirical findings in this study support the claim that the Accuracy of the classification job is much enhanced by promoting information exchange across agents in the MAS-DDM architecture. The combination of MAS and DDM offers a potential opportunity to enhance the capabilities of distributed data mining. It addresses concerns related to autonomy, privacy, and efficiency while also fostering a more flexible and collaborative data analysis environment.

This study makes the following noteworthy contributions to the field:

- Introduction of MAS-DDM: We propose MAS-DDM as an innovative solution, marrying multiagent systems with distributed data mining, thus overcoming the rigidity and limitations of traditional DDM approaches.
- Enhanced Adaptability: MAS-DDM introduces a communicative process among agents, enabling the exchange of knowledge, even when dealing with classified data. This enhances the adaptability and effectiveness of the overall system.

Section 2 of the study provides an overview of the pertinent research on the utilization of MAS and DDM. A detailed description of the proposed MDS system may be found in Section 3. The empirical results obtained from implementing and evaluating the proposed methodology are outlined in Section 4. The document is finalized in Section 5.

2. Related Work

Support Vector Machines (SVM) have been a prominent focus of machine learning research for an extended period. SVM training methods possess excellent theoretical qualities and have demonstrated high efficiency in numerous practical applications. However, the design of these algorithms frequently presents difficult optimization challenges. In this context, the proposal is to examine the fundamentals of Support Vector Machine learning through the standpoint of multiagent optimization. Multiagent systems decompose intricate optimization problems into basic "oracle" jobs and execute a cooperative solutions procedure, leading to a self-organized resolution of complicated problems. The authors demonstrate how the SVM training problem can be approached from this perspective and offer various insights into binary categorization, hyperparameter selection, multiclass learning, and unsupervised learning [11]. The conceptual work is exemplified using straightforward examples to elucidate the notions and comprehend the behavior of agent cooperation. The presented models offer concise formulations of intricate learning challenges that are occasionally arduous to complete using conventional optimization procedures. The principles being explored offer new possibilities for designing novel distributed cooperative learning systems.

This study presents a novel approach for credit risk assessment, utilizing a four-stage ensemble learning strategy based on Support Vector Machines (SVM) and multiagent systems [12]. In order to train and validate the model, the original dataset is divided into two distinct subsets: the training set, which is used for training the model, and the testing set, which is used to evaluate the model's performance. Furthermore, diverse SVM learning approaches are employed to develop intelligent credit risk agents. In the third stage, multiple Support Vector Machine (SVM) agents are trained using subsets of training data and subsequently evaluated. The final stage involves aggregating the outcomes of all SVM agents into an ensemble result. An investigation and analysis are conducted to examine the generalization performance of the multiagent ensemble learning system based on Support Vector Machines (SVM), with a focus on the diversity of intelligent agents. The efficacy of the multiagent ensemble learning system, which utilizes Support Vector Machines (SVM), is evaluated using a dataset specifically designed for assessing the approval of business credit card applications.

Open wireless sensor networks and mobile ad-hoc networks in emergency services, combat, and health monitoring represent a concern of cyber risks, intrusions, and assaults. Intrusion detection, which distinguishes abuse from atypical activity to maintain secure, dependable network operations and services, is a major research challenge in network safety. Advanced computers and multiagent systems provide the finest intrusion detection. Soft computing and machine learning computational intelligence techniques have been used to create Intrusion Detection and Prevention Systems (IDPS) [13]. Still, there are no state-of-the-art reviews on their performance and effects on wireless environment intrusion recognition issues in cloud computing. This work reviews and classifies IDPS systems using classical artificial computational intelligence with multiagent assistance. In this study, the significance of the techniques and methodologies, their performance, and their limitations are analyzed. The limitations are addressed as challenges to obtaining IDPS requirements for a collaborative-based wireless IDPS (Co-WIDPS) architectural design. To identify threats more accurately, it combines fuzzy reinforcement learning knowledge management with a better technology platform. Finally, we discuss many prospective research issues that might advance computational intelligence-based Co-WIDPSs.

An adaptive intrusion detection system that can identify unexpected threats in real-time network data is crucial. Retraining conventional adaptive intrusion detection systems with known and new assaults requires a lot of computing resources and time. The Real-Time Multiagent System for an Adaptive Intrusion Detection System RTMAS-AIDS is suggested in this paper to enable the intrusion detection system to react to unforeseen threats in real time [14]. This approach detects typical activity and known assaults using multi-level hybrid SVM and ELM. An adaptive SVM model with parallel processes and MAS distribution detects and learns new assaults in real-time. Results demonstrate that the suggested technique greatly decreased unknown attack detection training costs compared to the current method. RTMAS-AIDS can identify Probe, R2L, and U2R assaults better than the non-retrained multiagent system utilizing the multi-level hybrid SVM and ELM models, according to the popular KDDCup'99 dataset. RTMAS-AIDS detected and learned unknown assaults quicker (up to 61%) than MAS-MLSE and MLSE and had a 95.86% detection accuracy.

This study suggests detecting network attacks using flow calculation and deep learning [15]. A real-time detection technique based on flow calculations and common patterns and a classification algorithm based on DBN-SVM makes up the approach. DBN-SVM improves classification accuracy while sliding window (SW) stream data processing allows real-time detection. Finally, a system is built to test the procedure. The CICIDS2017 open-source data collection is used to perform comparison experiments. The real-time detection efficiency of the technique surpasses existing machine learning methods. Attack classification accuracy is 0.7 percentage points greater than DBN and two percentage points higher than combined algorithm boosting and bagging. Thus, it can identify high-speed network breaches in real-time.

Multiagent systems (MAS) like intelligent transport systems, efficient power grids, and sensor networks for data collecting and analysis work together. Intelligent agents and MAS need decision-making to do more complicated tasks. This overview covers cooperative MAS decision-making models from the previous five years, including Markov decision processes, game theory, swarm intelligence, and graph theoretical models [16]. They examine reinforcement learning, dynamic programming, evolutionary computing, and neural networks that produce optimum and suboptimal policies. These concepts are also used in robotics, wireless sensor networks, cognitive radio networks, intelligent transport systems, and smart electric grids. We also define crucial terminologies and highlight remaining issues, such as bringing large data into decision-making, building autonomous, scalable, and computationally efficient algorithms, handling increasingly complicated tasks, and standardizing assessment criteria. Recent surveys have covered this area, but we examine relevant models and applications.

3. The proposed MDS method

The core paradigm Multiagent System for Distributed Data Mining using Support Vector Machines (MDS) allows geographically scattered agents to cooperate in classification. The technique utilizes many agents, each simulating a distributed location, with a unique dataset and prior assumptions regarding instances that may vary by site. Each agent produces its categorization form by creating a unique learning model and connecting directly to numerous sites. The agent decides whether to visit additional sites based on its learned model and site information.

MDS allows collaborative categorization without revealing raw data, which is important in sensitive applications. Rather, agents communicate categorization information while protecting their data. The method improves classification accuracy and allows dispersed agents to collaborate and share knowledge. The MDS approach has shown promise in bioinformatics, remote sensing, and online content categorization.

MDS is an efficient solution to distributed data classification in a collaborative setting. It is applicable in healthcare, finance, and transportation, where communication between sent sites is critical for decision-making.

Effective distributed data mining requires the ability to categorize uncommon instances that no agent can classify locally. In such cases, multi-location agents must cooperate. Most local agents calculate the instance's likelihood and categorize it individually. However, this independence is only maintained if the classification output's likelihood exceeds a "p" probability threshold. Only if the output classification probability exceeds the threshold is the local agent's classification model suitable to classify the input instance.

The local agent will seek support from other agents if the classification output probability is below "p". The other agents will help determine the case's likelihood and notify the asking agent of their judgments or beliefs, which are the calculated categorization. If the likelihood of the classification output is greater than "p," the agents will exchange the benefits of adopting these beliefs in instance classification and revise their hypotheses about each class of data. This collaborative decision-making method helps agents employ the network's knowledge and expertise more effectively, improving categorization outcomes.

After receiving the cooperating agents' findings, the requesting agent evaluates the class labels using a Support Vector Machine (SVM) approach. This research will detail class label accuracy. Figure 1 illustrates the necessity of collaboration in multiagent systems for effectively categorizing distributed input.

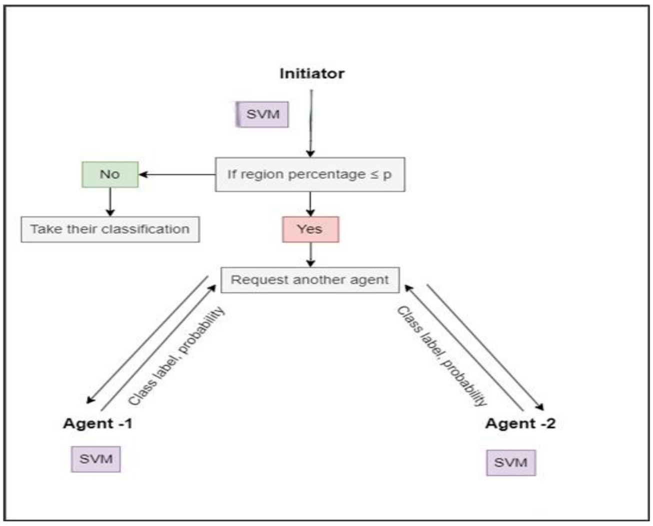


Figure 1. Proposed Framework of Collaborative Classification

The proposed work implements the paradigm for collaborative categorization via four distinct portions. The first stage entails the compilation of the dataset, including the gathering and refinement of data from many decentralized sources. During the second stage, each agent uses the Support Vector Machine (SVM) classification technique to construct their model using their dataset. The Support Vector Machine (SVM) algorithm offers an effective method of classification that is especially well-suited for handling data with a large number of dimensions. During the third step, the initiator uses feedback processing to determine whether to accept the class label supplied by other agents. Assessing a new approach that is appropriate for Support Vector Machines (SVM) is a critical stage in confirming the Accuracy of the final classification outcome.

The protocols used by agents in their communications and interactions inside the Multiagent System (MAS) are specified in the fourth phase. These protocols provide efficient communication and collaboration amongst agents, fostering the exchange of information and integrating diverse models to enhance the Accuracy of categorization. In summary, the suggested architecture offers a strong method for collaborative categorization. It allows agents in many locations to collaborate, using their unique skills and datasets to obtain exceptional classification results successfully.

3.1. Dataset Preparation

This study looks at cancer institutes around the globe, each with its unique patient case dataset, preconceptions, and diagnosis. These datasets may not apply to every area, culture, or organization. Conventional scenarios could need many cancer datasets from various sources for the same disease. This strategy is unsustainable due to the lack of datasets in publicly accessible web repositories. In order to overcome this difficulty, the dataset is distributed using k-means clustering, which also determines the appropriate number of agents for good cooperation. One popular clustering technique that determines the cluster number is the k-means algorithm.

Each agent chooses features from its cluster once the dataset has been clustered. Selecting the attributes that have the most influence on the prediction variable or outcome, either manually or automatically, is the most important step in the feature selection process. Inappropriate dataset features may reduce model accuracy and lead to model learning from unrelated data. Therefore, it is important to pick features once agents are assigned to clusters. Since each agent's qualities may vary based on their location and healthcare practices, the datasets are vertically scattered.

Using the same dataset and selecting characteristics based on their needs is ensured by this strategy. As a result, agent collaboration is made possible, and classification accuracy increases. The effectiveness of the suggested framework for collaborative categorization in dispersed scenarios with various datasets and limitations is shown by this case study.

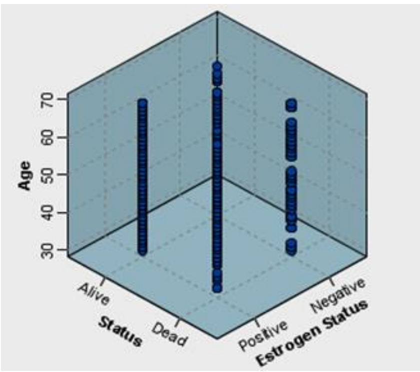
The dataset was subjected to clustering using the K-means method to enhance inter-agent communication and optimize the performance of the proposed approach. Experimenting with different cluster sizes showed that three clusters were the optimal choice for the dataset, considering the efficient communication capacity among agents. Due to the absence of any overlap among the agents in the examination of additional clusters, numerous attributes were insignificant and were thus lost throughout the feature selection process.

Feature selection was conducted on each cluster of the dataset, which had been previously grouped using the K-means technique. The objective of this method was to ascertain the most salient characteristics of each agent. Assessing the significance of each input in relation to the selected aim is a crucial step. The findings obtained for numerous agents are shown in Figure 7, illustrating the conclusion of this method.

3.2. SVM Classification Algorithm

This paper examines cancer hospitals worldwide that treat hard-to-classify patients. The suggested MAS-DDM strategy leverages SVM classification to tackle this problem. According to [17,18], the Support Vector Machine (SVM) algorithm is one of the best ways to address classification ambiguity.

Plotting data helps you understand and identify gaps. The graphic in Figure 2 shows data ambiguity. This stresses the need for an effective strategy to handle uncertain categorization cases, like the MDS method in this study.



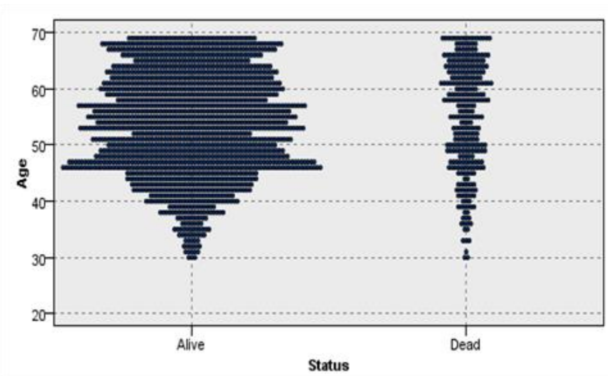


Figure 2 illustrates a graphical representation of the dataset utilized in the research.

The plot showcases the distribution of data points across relevant dimensions, providing a visual insight into the characteristics of the dataset. Different classes or categories may be differentiated by varying colors or markers, contributing to a clearer understanding of the dataset's structure. This visual depiction aids in discerning patterns, potential clusters, or trends within the dataset, laying the foundation for subsequent analyses and highlighting the significance of the dataset in the context of the research objectives.

One of the finest classification techniques for ambiguous data is the Support Vector Machine (SVM) [19]. SVMs are useful in uncertain situations because they can handle non-linearly separable data using kernel functions [20]. Even with noise and other uncertainties, SVM is still able to capture data structures.

The classification method known as Support Vector Machine (SVM) is renowned for increasing model prediction accuracy without overfitting training data. As a result, it is often used in fields where there are typically several predictor variables, such as bioinformatics, text mining, and image identification. Medical diagnostics, customer relationship management, and intrusion detection may all benefit from SVM [21].

To differentiate between examples with different class labels, SVM, a pioneering classifier, constructs hyperplanes in multidimensional space [22]. The SVM algorithm's classification of non-linearly separable data in a high-dimensional feature space is seen in Figure 3. After locating a category separator, the computer creates a hyperplane out of the data. The group of a new data point may then be predicted by the hyperplane using its attributes. Because many predictor fields must be avoided, SVM is helpful in bioinformatics, text mining concept extraction, and image identification [23]. Medical diagnostics, intrusion detection, and customer relationship management are more applications for SVM [24].

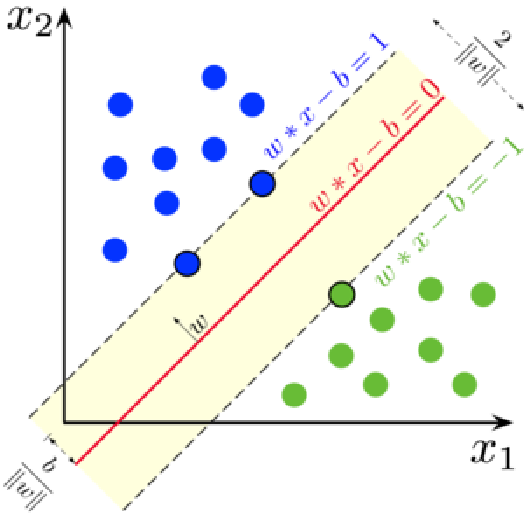


Figure 3 visually portrays the architecture of a Classic Support Vector Machine (SVM), a fundamental component in the research methodology.

The graphic shows the main components of an SVM: the decision border, support vectors, and the margin. The visualization may contain positive and negative examples to show how the SVM effectively classifies data points. This image illustrates Classic SVM concepts to help readers understand their function in the study's classification procedure.

In conclusion, support vector machines (SVM) are very useful classification methods that are able to handle ambiguous data. Through its ability to handle non-linearly separable data through the use of kernel functions, support vector machines (SVM) prove to be particularly useful in situations where there is a great deal of uncertainty. Image identification, bioinformatics, and medical diagnostics are just some of the numerous fields that might benefit from the use of support vector machines (SVM). Because of their ability to construct hyperplanes in a high-dimensional space and accurately predict the class label of incoming data points, support vector machines (SVM) are an excellent choice for classification tasks.

Support Vector Machines (SVM) are powerful classification algorithms that split cases with a variety of class labels by generating hyperplanes in a multidimensional space. It is necessary to use an iterative training method in order to minimize the error function and get an ideal hyperplane support vector machine. A support vector machine (SVM) is partitioned into two model groups for classification and two model groups for regression in accordance with the form of the error function. The following Equation illustrates how the training of a classifier of this sort is carried out by minimizing the error function while being required to adhere to certain constraints.

$$\frac{1}{2}\mathrm{d^t d} + \mathrm{A}\sum_{\mathrm{i=1}}^{\mathrm{N}} \quad \xi_\mathrm{i}$$

$$\mathrm{y_i(d^t \emptyset(s_i)} + \mathrm{c} \geq 1 - \xi_\mathrm{i} \mathrm{and} \xi_\mathrm{i} \geq 0, \mathrm{i} = 1, \dots, \mathrm{N}$$

Where the tiny constant is represented by c, the vector coefficient by d, the capacity constant by A, and the parameter to handle no separable input data is represented by ξ-i. The index of the$N$ training cases is represented by i. The independent variable is denoted by, while the class label is represented by the value (y(∈±1)). The input data is converted from an independent input space into a feature space using the kernel $\emptyset$. A penalized parameter will be shown as an enlarging A, so A should be carefully chosen to prevent overfitting.

The information above indicates that Support Vector Machines (SVMs) are an effective classification method with a wide range of applications, especially in unpredictable contexts. They can handle non-linearly separable data and capture its structure even in the presence of noise and other uncertainties because of kernel functions. Because Support Vector Machines (SVMs) optimize model prediction accuracy without overfitting the training data, they are able to assess large data sets containing hundreds of predictor variables.

Customer relationship management, voice and speech recognition, bioinformatics, text mining idea extraction, intrusion detection, face and image recognition, and medicine are among the fields in which support vector machines (SVMs) are helpful. As a result, SVMs are robust and flexible machine-learning techniques that can handle challenging classification problems, especially in unclear situations. Because they can adapt to high-dimensional feature spaces and handle non-linearly separable data, they are perfect for a wide range of applications.

3.3. Feedback Processing

The study suggests two methods to help the initiator accept the class labels of other agents. These techniques enhance information dependability and ultimate judgment. Below is an explanation of these techniques.

Each participating agent whose classification output meets a probability threshold "p" gives the querying agent a class label in the proposed collaborative classification architecture. This criterion ensures that class labels come from agents who are confident in their categorization outcomes. The requesting agency classifies using the most-voted class designation. Figure 4 shows the voting method, sometimes known as this strategy.

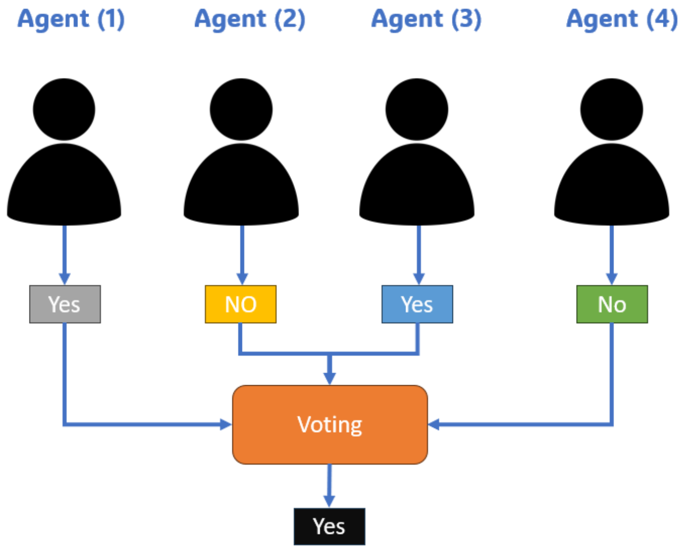


Figure 4 visually depicts the implementation of the Voting Method.

The picture shows several autonomous agents classifying things according to their perspectives. Arrows or connected lines show agent votes. Each agent's contributions might be colored or marked to emphasize the range of perspectives in the collaborative decision framework. Agents vote collectively in this figure, demonstrating the MAS's collective intelligence for proper classification.

The voting method makes it easy and efficient to integrate classification outputs from many agents by letting the requesting agent utilize the pooled knowledge of all participating agents to make a classification choice. When categorization results are uncertain, the approach helps the requesting agent make a better decision based on the participating agents' agreement. Overall, the voting strategy is a straightforward way to integrate classification results from several agents and has been employed in numerous applications. Combining numerous agents' skills in this simple way improves classification accuracy and reliability.

After requesting a label, the initiator agent receives several class labels with associated probabilities from other agents. The agent's Probability reflects the confidence of their class label based on their model and dataset features.

The initiating agent may select the data class label by voting based on likelihood. Out of all the labels supplied, the beginning agent of this technique selects the class label with the highest Probability. This method selects the label with the highest overall confidence by analyzing each agent's level of confidence in the provided label. Voting based on likelihood is shown in Figure 5. Each participant agent provides the class names and Probability of the starting agent. Lastly, given the given data, it selects the label with the highest Probability.

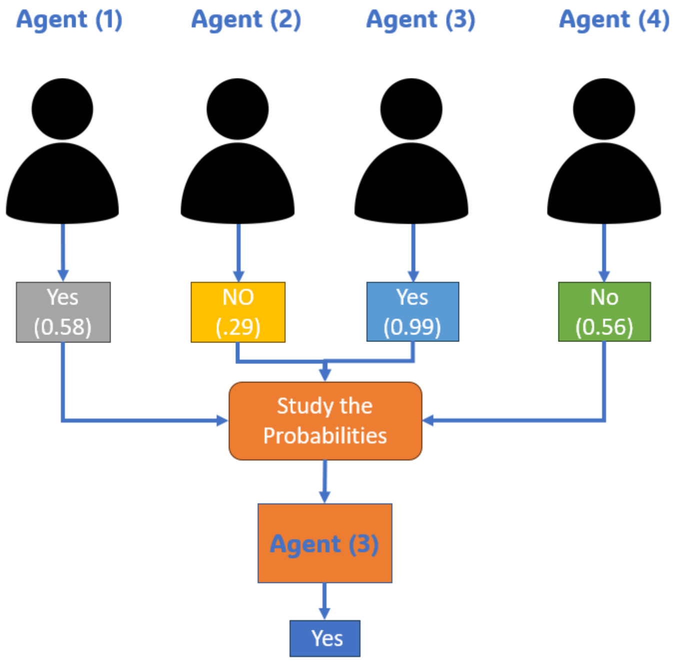


Figure 5. High Probability new method

When everything is taken into consideration, the probability-based voting strategy could be useful in situations where numerous agents may have varying degrees of expertise or access to different datasets, which might lead to differences in the labels that are assigned. The method selects the label that has the highest overall Probability as the final label, taking into consideration the amount of faith that each agent has in the label that they have been assigned.

For the purpose of selecting the right class label from among the several agents, the MAS-DDM system employs two different ways. The first approach is the voting process, which involves one agent asking other agents to submit class labels if they were able to obtain a classification result that satisfied a particular probability. "p." As a result, the class label that is used the most often is selected during the voting process. Figure 4 illustrates this method in its entirety.

In certain instances, the MAS-DDM voting procedure may create several class labels that have the same amount of votes, which will result in a tie. For the purpose of determining which class label is suitable to accept under these conditions, it is required to combine the two procedures. In order to find a solution to this problem, every agent that is now being contacted gives the agent that initiated the contact its class label likelihood. The initiator agent will choose the alternative that has a better possibility of occurring after assessing the many options. The conclusion is more consistent as a result of this technique, and the likelihood that the final class classification will be endorsed is increased.

The second way of the MAS-DDM system provide a more educated decision-making approach. This method takes into consideration the certainty of other agents' class label classifications based on their models, which results in a more informed approach to decision-making. Using this method, the initiator agent examines the replies and chooses the class label that has the greatest Probability after getting the class label probabilities from each of the agents that were contacted. The system is able to evaluate the degree of Accuracy with which each agent forecasts based on their model and arrive at a more educated judgment as a result of the use of this strategy. This tactic is shown in Figure 5, which is a visual illustration of the method. In general, the MAS-DDM system combines these two methods to ensure that the requesting agent is able to pick which class label to accept from other knowledgeable agents, even in circumstances when there is uncertainty or ambiguity.

3.4. Maximal Aerobic Speed Protocols

The present phase of the MDS technique [37] continues to implement the MAS protocol that was used in previous studies. Multiagent systems (MAS) use communication protocols to regulate the interactions and exchanges of information among agents. The Foundation for Intelligent Physical Agents (FIPA) is a consortium dedicated to establishing protocols and guidelines for the advancement of intelligent agents and multiagent systems while also promoting their widespread use. Agents may exchange messages using Agent Communication Language (ACL). FIPA protocols, including FIPA query IP, let one agent request another agent to do a certain task.

By using the FIPA-developed query IP protocols, an initiator agent may request other agents to perform tasks. These criteria specify the kind of communication, its format, and the circumstances and states being exchanged. The communication in question employs messages that fall into two categories: requests or calls, as specified by the Agent Communication Language (ACL). The ACL also describes the message's structure. The protocol provides a detailed account of the terms of the transaction.

A FIPA ACL message consists of one or more message parameters, and the particular requirements for these elements may vary depending on the nature of the agent communication. ACL messages often include parameters for the sender, recipient, and content. However, the performative parameter is mandatory for all ACL messages.

The MDS technique employs many agents, each with its distinct dataset. These agents autonomously perform activities while concurrently communicating and receiving information from other agents. Agents may communicate with one other by using FIPA query IP and ACL messages. These messages provide specific requests or responses for certain types of information. The use of these communication protocols across a wide range of multiagent development environments has shown to be advantageous for MAS implementations.

4. Experiments and Results

The IBM® SPSS® Modeler software, which was coupled with Python and the Smart Python Multiagent Development Environment (SPADE), was used in order to evaluate the effectiveness of the collaborative categorization strategy that was described in this research. In order to determine whether or not the method was effective, a dataset that was collected via the IEEE data port was used. Data from the SEER Program of the National Cancer Institute's November 2017 update were included in this dataset, which included information on breast cancer patients. These cancer statistics are derived from the whole population and are provided by this program. The dataset, which consisted of forty-two hundred and twenty-four occurrences and nine attributes, was created with the intention of predicting the survival rate of cancer patients. In order to classify the qualities of the class label, we divided them into two categories: living and dead.

Python, SPADE, and the IBM® SPSS® Modeler software were among the open-source resources that were used in the development of the method of categorization (MAS). Additionally, the SPADE framework makes it easier to create, implement, and maintain Multiagent Systems (MAS) in situations that include networked computing.

The dataset that was used in this investigation is a dataset that is widely utilized for research on classification problems and is easily accessible to the general public. In light of the fact that the SEER Program of the National Cancer Institute (NCI) offers trustworthy and up-to-date information on cancer statistics at the population level, the use of a dataset is the best option for testing the suggested strategy.

The IBM® SPSS® Modeler software, Python, and SPADE were used in order to successfully construct and assess the Multiagent System (MAS) method that was presented for collaborative categorization. When evaluating the technique's overall performance, it is possible to take into consideration the precision of categorization as well as the effectiveness of the MAS in maintaining the data. An in-depth analysis of the results of the research is going to be presented in the next part.

The dataset is then divided among the agents, each of whom receives a unique training and testing set once the clustering and feature selection procedures are finished. Hence, each unique agent applies the SVM classifier to the data, as shown in Figure 6. The level of Accuracy attained by each agent using their models is shown in Table 1. Using this strategy, agents may work together to improve classification performance while training on their local data at the same time.

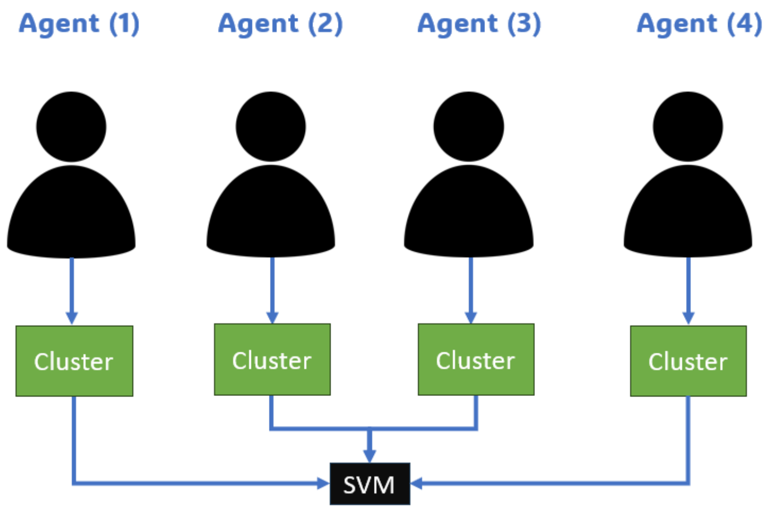


Figure 6 provides a visual representation of the distribution of data among autonomous agents within the Multiagent System (MAS)

Table 1. Accuracy of Each Agent's Model

Agents	Overall Accuracy
Agent-1	86.50
Agent-2	89.65
Agent-3	92.98
Agent-4	95.10

A multiagent system (MAS) for categorization assignment was built using the SPADE platform, a development of the fully featured FIPA (Foundation for Intelligent Physical Agents) platform. SPADE agents may access predetermined actions via repeating patterns. These behavior categories allow agents to execute many tasks and accomplish many functions. Every behavior has documented states and transitions. This behavior design lets SPADE agents develop complicated agent model behaviors. With SPADE, the MAS in this study completed the classification assignment by exchanging data and enabling agents to learn from their local data while working together to improve classification performance.

In the MDS approach that has been proposed, choosing the optimal decision threshold for the classifier is necessary in order to reduce the likelihood of erroneous detection. The decision threshold that results in the model having the lowest possible likelihood of misclassification was found via a series of iterations in order to accomplish this goal.

Change the decision threshold and evaluate the performance of the classifier using metrics such as Accuracy, precision, recall, and F1-score. This will allow you to accomplish the desired result. In the next step, the decision-making threshold that yields the best possible outcomes is selected.

In addition, the effectiveness of the MDS methodology was evaluated in comparison to that of the conventional DDM method, which is used for categorization by each agent in a manner that is independent of the others. A comparison was made between the models that were produced by each approach, taking into consideration their Accuracy, precision, recall, and F1 score. In terms of classification accuracy and F1 score, the findings demonstrated that the MDS strategy performed better than the traditional DDM.

Table 2. Overall Results

Agent-1			
Test #	With Collaboration		Without Collaboration
	Method 1	Method 2	
1	81.45	81.14	79.57
2	91.24	88.53	72.58
3	93.15	90.38	70.68
4	94.54	91.47	70.17
Agent-2			
Test #	With Collaboration		Without Collaboration
	Method 1	Method 2	
1	92.14	91.47	67.00
2	95.88	94.19	67.33
3	96.36	96.75	67.75
4	97.99	97.74	69.54
Agent-3			
Test #	With Collaboration		Without Collaboration
	Method 1	Method 2	
1	91.74	91.16	69.59
2	90.97	90.64	72.64
3	93.69	91.96	71.69
4	94.47	92.17	72.33
Agent-4			
Test #	With Collaboration		Without Collaboration
	Method 1	Method 2	
1	91.66	90.12	70.52
2	86.47	84.48	73.98
3	92.68	92.26	70.59
4	93.42	93.37	70.98

As a consequence of this finding, it seems that the proposed approach prioritizes individual agents' local learning while allowing for the potential of agent collaboration when needed. It is not required for an agent to ask other agents for information if that agent is capable of securely classifying an instance by itself. On the other hand, when an instance is more challenging to classify or when one agent is unsure about the classification of the instance, working with other agents may improve the overall classification accuracy. This approach achieves a balance between the benefits of cooperating in a distributed system and local learning. Testing a model on a subset of the dataset is a common method in the machine learning field to evaluate the model's performance. In this case, twenty percent of the dataset used to assess each agent comprised occurrences without a class designation. This meant that the model had to create assumptions about the class labels based on the characteristics that were supplied rather than being given the actual class names. This approach may help determine if the model is right and can also be used to identify potential issues like under- or overfitting. If each agent were tested separately, the effectiveness of the proposed method could be evaluated per agent. This would make it possible to identify and solve any issues that could develop with certain agents.

Each agent went through the process of building a model before implementing the suggested system to confirm the communication and process flow. The findings demonstrated that, in terms of Accuracy, the MDS technique outperformed the traditional DDM algorithm. Tables 2 and Figures 9–11, which are connected to each agent, provide a more thorough presentation of these results. Overall, the results demonstrated the effectiveness of the proposed MDS technique in raising classification performance and lowering the Probability of false positives in cancer datasets.

It is important to point out that the voting mechanism for the MDS strategy that was proposed was shown to be the most effective for all of the testing sets being considered. When compared to the first way, technique two, which has a high likelihood, provided a considerable improvement for agent two in the third testing set. The use of agent collaboration increases the system's level of Accuracy. According to the statistics, the Accuracy of the MDS approach that was recommended is greater than that of the system that does not include cooperation. The Accuracy of the MDS technique increased from 18.74% to 26.02%. As a consequence of these findings, the importance of collaboration in improving classification performance and reducing the likelihood of producing false positives in cancer datasets is clearly shown.

Table 3 is presented here in order to provide the outcomes of the experiment in a manner that is both thorough and easy to understand. The findings unequivocally demonstrate the advantages of the MDS method that was presented in terms of enhancing the overall performance of the system by enhancing the classification accuracy of the agents.

Table 3. Overall Accuracy of the given methods

Overall Accuracy			
Test #	With Collaboration		Without Collaboration
	Method 1	Method 2	
1	90.89	88.34	69.16
2	91.69	91.69	70.89
3	93.35	92.85	71.36
4	94.52	93.59	72.58

The experimental results obtained from the proposed MDS have shown that this approach consistently produces higher Accuracy and better results when compared to the non-cooperative system. It has also been shown that the two procedures used in the suggested approach enable the initiator agent to effectively decide whether or not to accept the class designation that other agents have provided to it. It is important to consider that every solution in the proposed technique yielded outcomes that were much superior to those generated by the non-collaboration system. These results demonstrate the significance of the mutual collaboration method in improving the contractual agents' performance, with an accuracy increase ranging from 18.74% to 26.02%.

5. Conclusion

In conclusion, the implementation of the proposed Mutual Collaboration Agent Classification using the Support Vector Machine (MDS) technique has been carried out in a Multiagent System (MAS) that has been built and placed into operation. Every agent has the ability to construct prior hypotheses based on unique data and preserve samples that are unique to itself using this method. In order to categorize freshly acquired data items, every agent employs the naïve Support Vector Machine approach. Both the shared information from other agents and the agent's own preconceived conceptions are used in this classification process. When agents working on classification tasks interact, they are motivated to provide knowledge that is both diverse and specific, depending on the data. This results in much-improved output. The most significant distinction between the MDS and the standard Support Vector Machine approaches is the manner in which the former makes use of probabilities. This is the case despite the fact that both techniques employ Probability to update and create classifier labels. The results of the experiments conducted on two different datasets demonstrate that the MDS approach that was presented is more accurate and generates better results than the system that does not include cooperation.

Furthermore, the findings indicate that MDS is efficient in terms of giving the initiator agent the choice to either accept or reject the class label that other agents have provided. In addition, the outcomes that MDS obtained were significantly superior to those provided by the non-collaboration system. When the model is constructed by making use of the properties of the dataset, it is very precise and efficient. This is based on the outcomes of the experiments. It would be interesting to conduct experiments with the MDS system using a variety of classifiers and for various applications. The purpose of these experiments would be to assess the adaptability of the proposed approach and its potential for use in a variety of sectors. Given all that has been taken into consideration, the MDS strategy that has been offered offers a promising option for further study into cooperative categorization systems.

Compliance with ethical standards

Conflict of Interest: The authors declare that there is no conflict of interest regarding the publication of this paper.

Ethical approval: This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent: Informed consent was obtained from all individual participants included in the study.

Data availability statements: Data is available from the authors upon reasonable request.

Acknowledgements: This work was partially funded by Middle East University.

Funding: Not Applicable

References

1. Zhang, N., & Bao, H. (2009, April). Research on distributed data mining technology based on Grid. In 2009 First International Workshop on Database Technology and Applications (pp. 440-443). IEEE.
2. Luo, P., Lü, K., Shi, Z., & He, Q. (2007). Distributed data mining in grid computing environments. Future Generation Computer Systems, 23(1), 84-91.
3. Denham, B., Pears, R., & Naeem, M. A. (2020). HDSM: A distributed data mining approach to classifying vertically distributed data streams. Knowledge-Based Systems, 189, 105114.
4. Ashrafi, M. Z., Taniar, D., & Smith, K. (2002). A data mining architecture for distributed environments. In Innovative Internet Computing Systems: Second International Workshop, IICS 2002 Kühlungsborn, Germany, June 20–22, 2002 Proceedings 2 (pp. 27-38). Springer Berlin Heidelberg.
5. da Silva, J. C., Giannella, C., Bhargava, R., Kargupta, H., & Klusch, M. (2005). Distributed data mining and agents. Engineering applications of artificial intelligence, 18(7), 791-807.
6. Devi, S. V. S. G. (2014). A survey on distributed data mining and its trends. International Journal of Research in Engineering & Technology (IMPACT: IJRET), 2(3), 107-120.
7. Tsoumakas, G., & Vlahavas, I. (2009). Distributed data mining. In Database Technologies: Concepts, Methodologies, Tools, and Applications (pp. 157-164). IGI Global.
8. Qasem, M. H., Obeid, N., Hudaib, A., Almaiah, M. A., Al-Zahrani, A., & Al-Khasawneh, A. (2021). Multiagent system combined with distributed data mining for mutual collaboration classification. IEEE Access, 9, 70531-70547.
9. Kamdar, R., Paliwal, P., & Kumar, Y. (2018). A state of art review on various aspects of multiagent system. Journal of Circuits, Systems and Computers, 27(11), 1830006.
10. Gonçalves, E. J. T., Cortés, M. I., Campos, G. A. L., Lopes, Y. S., Freire, E. S., da Silva, V. T., ... & De Oliveira, M. A. (2015). MAS-ML 2.0: Supporting the modelling of multiagent systems with different agent architectures. Journal of Systems and Software, 108, 77-109.
11. Couellan, N., Jan, S., Jorquera, T., & Georgé, J. P. (2015). Self-adaptive support vector machine: a multiagent optimization perspective. Expert systems with Applications, 42(9), 4284-4298.
12. Yu, L., Yue, W., Wang, S., & Lai, K. K. (2010). Support vector machine based multiagent ensemble learning for credit risk evaluation. Expert Systems with Applications, 37(2), 1351-1360.
13. Shamshirband, S., Anuar, N. B., Kiah, M. L. M., & Patel, A. (2013). An appraisal and design of a multiagent system based cooperative wireless intrusion detection computational intelligence technique. Engineering Applications of Artificial Intelligence, 26(9), 2105-2127.
14. Al-Yaseen, W. L., Othman, Z. A., & Nazri, M. Z. A. (2017). Real-time multiagent system for an adaptive intrusion detection system. Pattern Recognition Letters, 85, 56-64.
15. Zhang, H., Li, Y., Lv, Z., Sangaiah, A. K., & Huang, T. (2020). A real-time and ubiquitous network attack detection based on deep belief network and support vector machine. IEEE/CAA Journal of Automatica Sinica, 7(3), 790-799.
16. Rizk, Y., Awad, M., & Tunstel, E. W. (2018). Decision making in multiagent systems: A survey. IEEE Transactions on Cognitive and Developmental Systems, 10(3), 514-529.
17. Anggoro, D. A., & Novitaningrum, D. (2021). Comparison of accuracy level of support vector machine (SVM) and artificial neural network (ANN) algorithms in predicting diabetes mellitus disease. ICIC Express Letters, 15(1), 9-18.
18. Çomak, E., Arslan, A., & Türkoğlu, İ. (2007). A decision support system based on support vector machines for diagnosis of the heart valve diseases. Computers in biology and medicine, 37(1), 21-27.

19. Got, A., Zouache, D., Moussaoui, A., Abualigah, L., & Alsayat, A. (2023). Improved Manta Ray Foraging Optimizer-based SVM for Feature Selection Problems: A Medical Case Study. Journal of Bionic Engineering, 1-17.

20. Safaldin, M., Otair, M., & Abualigah, L. (2021). Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. Journal of ambient intelligence and humanized computing, 12, 1559-1576.

21. Abualigah, L., Ahmed, S. H., Almomani, M. H., Zitar, R. A., Alsoud, A. R., Abuhaija, B., ... & Elaziz, M. A. (2024). Modified Aquila Optimizer Feature Selection Approach and Support Vector Machine Classifier for Intrusion Detection System. Multimedia Tools and Applications, 1-27.

22. Al-Manaseer, H., Abualigah, L., Alsoud, A. R., Zitar, R. A., Ezugwu, A. E., & Jia, H. (2022). A novel big data classification technique for healthcare application using support vector machine, random forest and J48. In Classification applications with deep learning and machine learning technologies (pp. 205-215). Cham: Springer International Publishing.

23. Ghasemi, M., Zare, M., Zahedi, A., Trojovský, P., Abualigah, L., & Trojovská, E. (2024). Optimization based on performance of lungs in body: Lungs performance-based optimization (LPO). Computer Methods in Applied Mechanics and Engineering, 419, 116582.

24. Gharaibeh, H., Mamlook, A., Emhamed, R., Samara, G., Nasayreh, A., Smadi, S., ... & Abualigah, L. (2024). Arabic sentiment analysis of Monkeypox using deep neural network and optimized hyperparameters of machine learning algorithms. Social Network Analysis and Mining, 14(1), 1-18.