2026 Volume: 02 Issue: 09 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-12

DESIGN OF EFFICIENT DEEP LEARNING CLUSTERING FOR DATA MINING IN LARGE DATA SET

***C. Premila Rosy**

Assistant Professor and Head, PG and Research Department of Computer Science, Idhaya College for Women (Autonomous), Affiliated to Bharathidasan University, Kumbakonam.

Article Received: 05 August 2026
Article Revised: 25 August 2026
Published on: 15 September 2026

***Corresponding Author: C. Premila Rosy**
Assistant Professor and Head, PG and Research Department of Computer Science, Idhaya College for Women (Autonomous), Affiliated to Bharathidasan University, Kumbakonam.

DOI: https://doi-doi.org/101555/ijrpa.9474

ABSTRACT

Revolutionizing data analysis, data mining employs predefined procedures and algorithms to extract valuable patterns from datasets. Over the years, diverse approaches such as Frequent Pattern Mining, Association Rule Analysis, Classification, and Clustering have emerged as cornerstones in data mining. Clustering, for instance, focuses on grouping dataset elements such that those within a group exhibit greater similarity than those in other groups. This research is dedicated to enhancing partition-based clustering algorithms, with a specific focus on Deep SVM. The objective is to imbue these algorithms with advanced features to efficiently analyze diverse datasets comprising numerical, categorical, and mixed data. Real-world datasets, particularly those with high dimensionality, pose challenges to the efficiency of Deep SVM clustering, rendering it computationally intensive and economically prohibitive. Consequently, cluster quality diminishes as dataset sizes increase. The core aim of this research centers on the development of robust clustering algorithms. Clustering applications span a wide array of domains from identifying underperforming products within a company inventory to grouping students based on academic performance for targeted interventions. Deep SVM emerges as a user-friendly clustering algorithm adept at handling sizable, high-dimensional datasets, fostering quick and efficient analysis. However, certain limitations persist within this approach. The efficacy of the K-Modes technique encounters hurdles with expansive datasets characterized by elevated dimensionality. As dimensions proliferate, the computational demands of the K-Modes approach surge, resulting in compromised cluster quality. To mitigate these issues, enhancements are proposed through

the elimination of non-significant features from the clustering process. This tactful reduction in cluster dimensionality not only bolsters accuracy but also curtails computational overhead. Moreover, the proposed algorithm incorporates a dynamic parameter, influenced by user preferences. Notably, the input parameter remains effective when its significance value surpasses 60% of the maximum significance value.

KEYWORDS: Frequent Pattern Mining, Clustering Algorithms, Association Rule Analysis, computational efficiency.

1. INTRODUCTION

Using data mining, patterns and knowledge may be collected from the massive amounts of data collected by information systems, and these patterns can be used to make sound judgments. Organizations can make better decisions because of it. Data mining offers methods for efficiently processing massive amounts of data and presenting it in the desired format [1]. Data mining is used in Data Mining Techniques refers to the preset procedures and algorithms used to extract these valuable patterns. Frequent Pattern Mining, Association Rule Analysis, Classification, and Clustering are all common data mining approaches that have been developed over the years [2]-[4].

In order to extract patterns of knowledge implicitly stored in massive data sets, DM offers a practical solution. This is a critical step in the discovery process of new information. Advances in computing technology have increased this discipline ability to collect and process data. Today heterogeneous information sources and real-time nature of transactions pose a significant difficulty because of the massive volumes of organised and unstructured data that must be processed. It difficult to mine unstructured data, which is the most frequent type of information, but the capacity to do so will reveal important patterns [5]. Unstructured data presents a number of difficulties, including data variances and pattern recognition.

The speed of Deep SVM is what makes it popular. The approach has a low computational cost and performs well on datasets that are both high dimensional and huge. It does, however, have significant drawbacks [6]. There is a key drawback in that the clusters formed are very dependent on the initial centroids selected (cluster centers). Because the initial centroids are chosen at random, different runs of the Deep SVM algorithm on the same dataset may not provide the same result [7].

To overcome this obstacle, a great deal of effort has been expended. Other restrictions include needing to enter a known number of clusters as an input; in this case, it is K. When

the dataset is fresh or unknown, an erroneous number of clusters is required, resulting in wasteful data grouping. Researchers are continuously looking for ways to get around this obstacle. When working with categorical data, K-Mean constraints are carried over to its expansions K-Modes and K-Prototype.

High dimensional datasets also pose a significant barrier to K-Mode efficiency. Computationally expensive for K-Modes, which results in decreasing quality for larger cluster sizes. Researchers are looking for ways to get over K-Mode current limitations.

- To circumvent the limitation of supplying the number of clusters at the beginning of the method by extending the Deep SVM for numerical datasets algorithm.
- Using a modified version of the K-Modes approach, we can cluster categorical datasets without worrying about having to enter the initial number of clusters.
- For mixed datasets, adapt the K-Prototype algorithm to circumvent the initial cluster supply limitation.
- This comparison should be made on a variety of real-world datasets, ranging in size and complexity from small to large.

Methods provided in this paper yield clusters with more accuracy than the original Deep SVM, K-Modes, and K-Prototype algorithms, as well as many extensions proposed by different authors.

A complete business analytics workbench, RapidMiner is an open source system with an emphasis on data mining, text analysis and predictive analytics. Many descriptive and predictive techniques are used to offer you the information you need. With RapidMiner and RapidAnalytics, a complete business intelligence solution with predictive analytics is available with full reporting and dash boarding capabilities. Real datasets from UCI Machine Repository: a website that provides the machine learning community with 300 datasets are used in the experiments.

2. Literature Survey

The work in [8] addressed the initial centroids selection sensitivity problem within the Deep SVM algorithm. The Forgy method tackled this by randomly selecting starting centroids from the dataset. This approach leveraged the probability of choosing points near densely populated areas, enhancing the algorithm performance. However, this approach didn't guarantee that the chosen seeds would always be within the same cluster or even as outliers.

In [9], took a unique approach by assigning instances one by one in the order of appearance in the database. Subsequent Deep SVM iterations were then performed, refining cluster assignments. This method, although more precise, incurred computational complexity due to numerous iterations, particularly in large databases.

In [10], the algorithm emerged, introducing an innovative strategy for initial centroid selection. By choosing an edge object as the first centroid, followed by objects farthest from it, and subsequently selecting objects farthest from their nearest centroids aimed to enhance clustering accuracy. Nevertheless, the method effectiveness was compromised when outliers or noise were treated as seeds.

In [11] proposed a strategy to manage complex datasets by dividing them into subsets. Each subset underwent Deep SVM with Forgy-based initial centroids, and the results from these iterations were aggregated. However, this approach didn't guarantee consistent outcomes across the entire dataset.

In [12] harnessed the kd-tree data structure to address the computational intensity of the standard algorithm nearest-neighbor queries. This approach used kd-nodes tree statistics to optimize cluster center updates, although it struggled with large, multidimensional datasets.

Building upon these foundations, this research proposes innovative numerical, categorical, and mixed dataset techniques within the Deep SVM family. Addressing the efficiency and sensitivity challenges of Deep SVM clustering, particularly with high-dimensional real-world datasets, remains a focus. Although advancements have been made for numerical datasets, challenges persist in dealing with categorical and mixed datasets.

3. Proposed Method

The proposed method aims to enhance clustering algorithms, particularly Deep SVM, to efficiently analyze datasets of varying types, including numerical, categorical, and mixed data, while addressing the challenges posed by high dimensionality. The overarching goal is to improve the quality of clusters while maintaining computational efficiency.

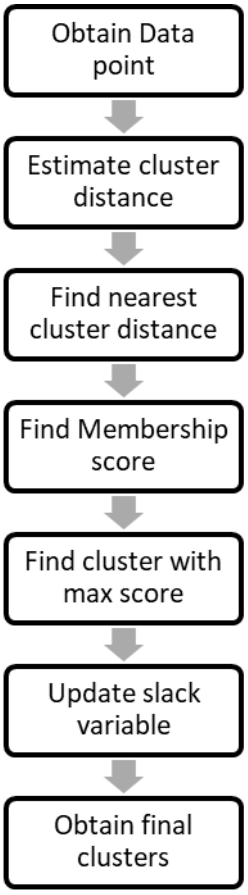

Figure 1: Proposed Method.

Deep SVM Algorithm

Deep SVM is a clustering algorithm that leverages the principles of Support Vector Machines for clustering tasks. It efficiently identifies clusters by finding decision boundaries that separate different data points. This algorithm is known for its speed and ease of use, making it suitable for large datasets with high dimensionality.While Deep SVM offers speed and effectiveness, it encounters difficulties when dealing with real-world datasets characterized by a large number of dimensions. As the dataset dimensions increase, the computational resources required to carry out the clustering process escalate, often resulting in reduced cluster quality and increased implementation costs.The goal is to find decision boundaries that separate different clusters of data points in a high-dimensional space.

In traditional SVM classification, the algorithm seeks a hyperplane that maximizes the margin between two classes of data points. The equation of the hyperplane can be represented as:

$Wx+b = 0$

where:

w: Weight vector perpendicular to the hyperplane.

x: Input data vector.

b: Bias term.

In real-world data, it is common to have some amount of noise or overlapping between classes. Soft Margin SVM introduces slack variables (ξ) to allow some misclassification. The objective becomes a trade-off between maximizing the margin and minimizing misclassification:

$wx+b \geq 1 - \xi i$

$wx+b \leq \xi i - 1$

where:ξi: Slack variable for the i-th data point.

Deep SVM extends SVM to clustering by aiming to separate clusters in the feature space. It uses a similar decision boundary concept, but instead of classes, it aims to find decision boundaries between clusters. The algorithm iteratively adjusts these boundaries to optimize cluster separation.

$\sum_{xj}\sum_{i}\xi i \geq$ threshold

where:

xj: Data point in cluster i.

ξj: Slack variable for data point xj.

The optimization problem involves finding the weights w and bias b that minimize misclassifications while maximizing the margin or cluster separation.

Algorithm: Deep Support Vector Machine (Deep SVM) Clustering

1. Input:
   - Dataset with data points X of shape (nsamples, nfeatures).
   - Number of clusters K.
   - Maximum number of iterations maxiter.
   - Slack variable threshold threshold.
2. Initialization:
   - Initialize cluster centers C.
   - Initialize slack variables ξ for each data point to zero.
3. Iterative Optimization:
   - Repeat for iter from 1 to maxiter:
     - For each data point xi:
       - Calculate the distance between xi and all cluster centers in C.

- Assign xi to the nearest cluster center and update the cluster assignment.
- For each cluster k:
- Update the cluster center ck as the mean of all data points assigned to cluster k.
- For each data point xj:
- Calculate the cluster membership score for each cluster k using the distance to the corresponding cluster center.
- Identify the cluster kmin with the maximum membership score.
- Update the slack variable ξj based on the difference between the membership score of cluster kmin and the membership scores of other clusters.
- Check if the sum of slack variables for each cluster is less than the threshold. If yes, stop iterations.

4. Output:
- Cluster assignments for each data point.

Proposed K-Modes Incorporation:

To address the limitations of Deep SVM in handling high dimensionality and maintaining cluster quality, the proposed technique integrates concepts from the K-Modes algorithm, which is recognized for its ability to cluster categorical data effectively.The proposed technique involves preprocessing the dataset to reduce dimensionality and enhance cluster quality. Non-significant features are identified and removed from the clustering process. This strategic reduction of cluster dimensions not only mitigates computational complexity but also enhances the accuracy of the resulting clusters.The proposed method introduces a dynamic parameter that plays a pivotal role in optimizing the clustering process. Users have the flexibility to influence this parameter based on their specific requirements. The parameter significance value is evaluated and compared against a predefined threshold (60% of the maximum significance value), ensuring its relevance and impact on the clustering outcomes.By integrating the strengths of Deep SVM with the effective clustering techniques of K-Modes, the proposed method strikes a balance between computational efficiency and cluster quality. It addresses the challenges associated with large dimension datasets, enabling the analysis of complex real-world data while yielding meaningful and accurate clusters.

4. RESULTS AND DISCUSSION

In the analyses, the proposed technique has been evaluated across a range of datasets, including the Post-Operative Patient dataset from the KEEL Data Repository. The comparison involves assessing the performance of the proposed algorithm against the original K-Prototype algorithm and various algorithms from the existing literature. The RapidMiner K-Prototype algorithm employs a predetermined initial value **K** for clustering known data.

The dataset under consideration comprises 8 attributes, with 7 of them being categorical and

1 numerical. A total of 90 items are categorized into 3 clusters. Typically, the data exhibits a division into two groups of roughly equal sizes: one with 24 items and the other with 64 items. However, due to three missing records, the experiment considers a total of 87 items. The results of the dataset analysis are presented in Table 1, which offers a comparison between the outcomes of the proposed Deep SVM algorithm and the K-Prototype algorithm. For the K-Prototype algorithm, the input **K** value is set at 3.

Table 1: Post-Operative Patient Dataset.

Cluster No.	**K-Prototype Algorithm**	**Proposed Deep SVM**
	1 2 3	1 2 3-10
Matching Categorical Attributes	2 1 0	2 1 2-8
Records per Cluster	4 64 19	15 55 1-4

From the table 1, it is apparent that the suggested algorithm generates ten clusters, with only two of these clusters being significant. Out of the total 87 items, these two significant clusters encompass 70 items. The remaining 17 items are distributed among clusters of sizes ranging from 3 to 10. These clusters exhibit two categorical attributes with matching values and a numerical attribute with values of 10 or 15. Cluster 1 consists of 15 items that share a single categorical attribute and a numerical attribute with values of 10 or 15. Cluster 2 encompasses 55 items sharing a categorical attribute and a numerical attribute with values of 5 or 10. Minor clusters comprise a varying number of items from three to ten. The proposed algorithm's clusters within this range contain 2 to 4 items each, sharing similarities in at least two categorical attributes and a numerical attribute with closely valued attributes.

Comparing the proposed algorithm's performance to the K-Prototype algorithm, the proposed algorithm is better at grouping items with the highest similarity. Unlike the proposed approach, the K-Prototype algorithm does not create clusters containing items that are most similar to each other.Furthermore, the accuracy of the proposed algorithm exceeds that of most other algorithms, as demonstrated in figure 2, which presents the accuracy of various clustering algorithms:

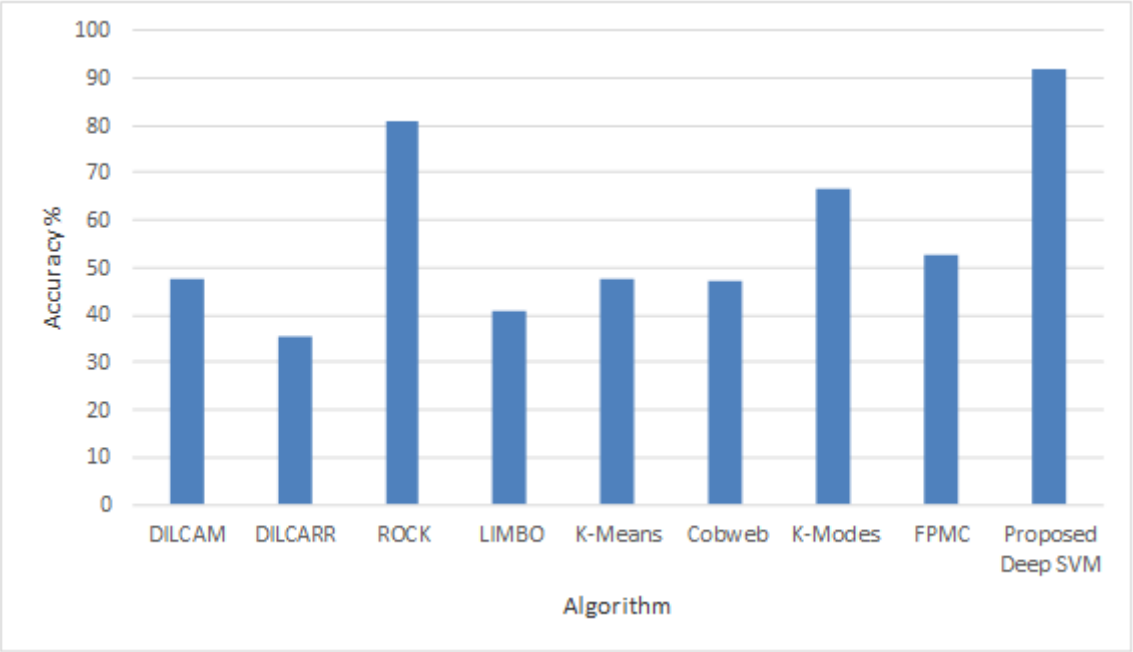

Figure 2: Accuracy of Clustering Algorithms.

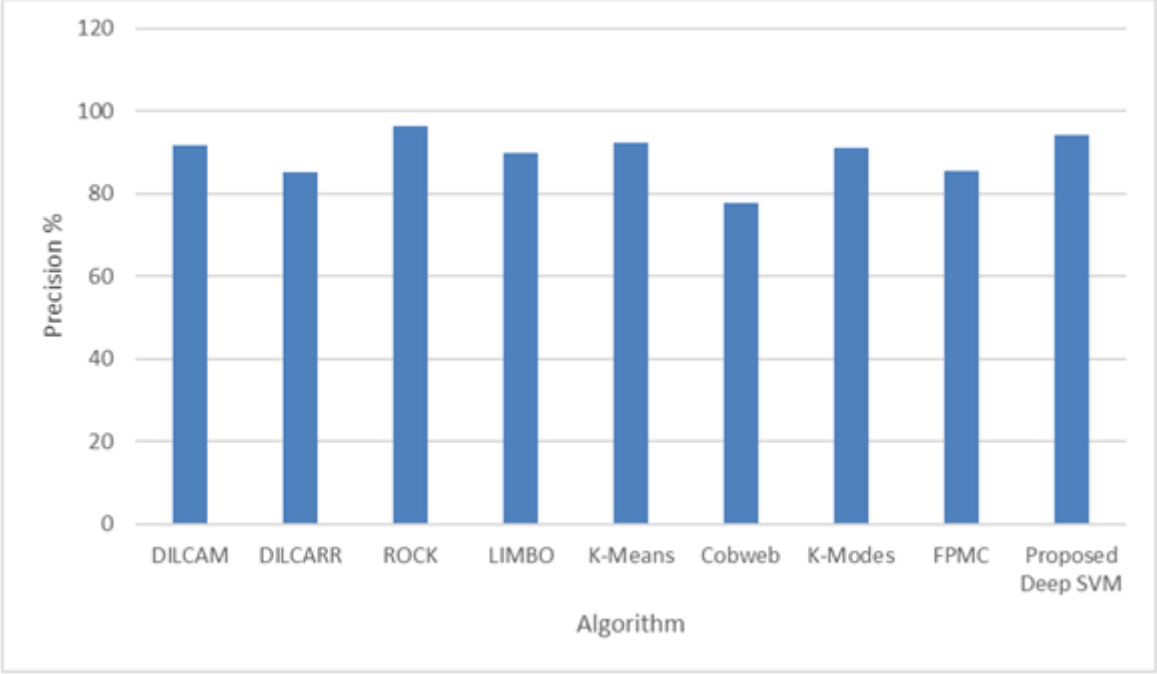

Figure 3: Precision of Clustering Algorithms.

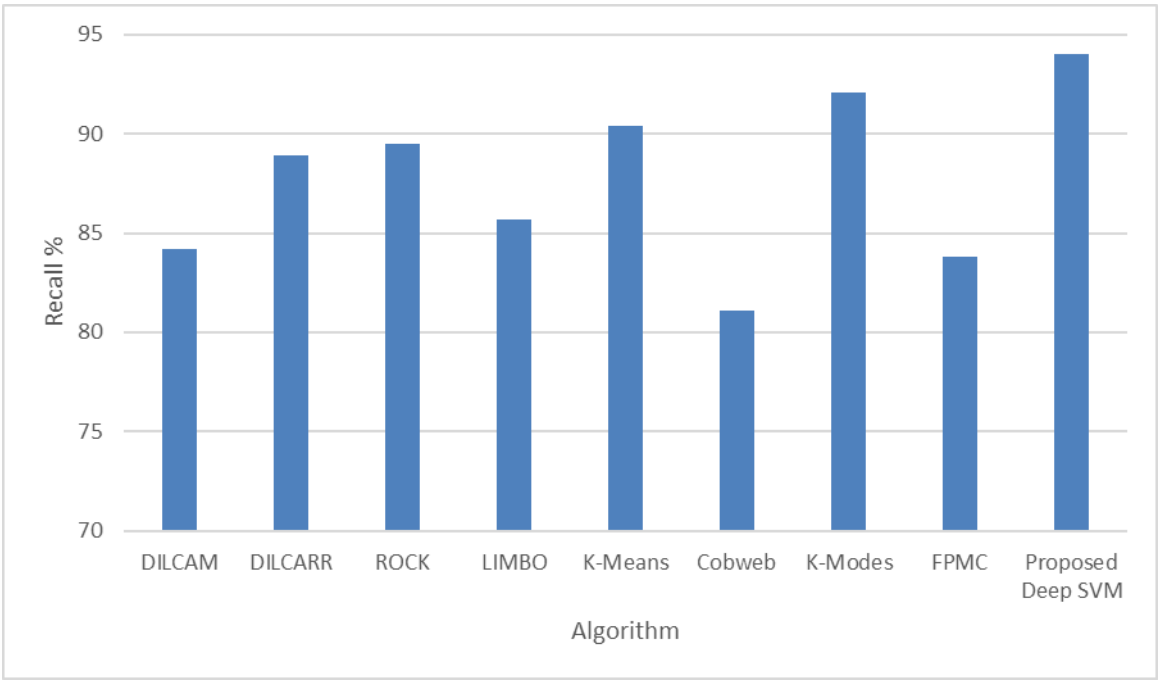

Figure 4: Recall of Clustering Algorithms.

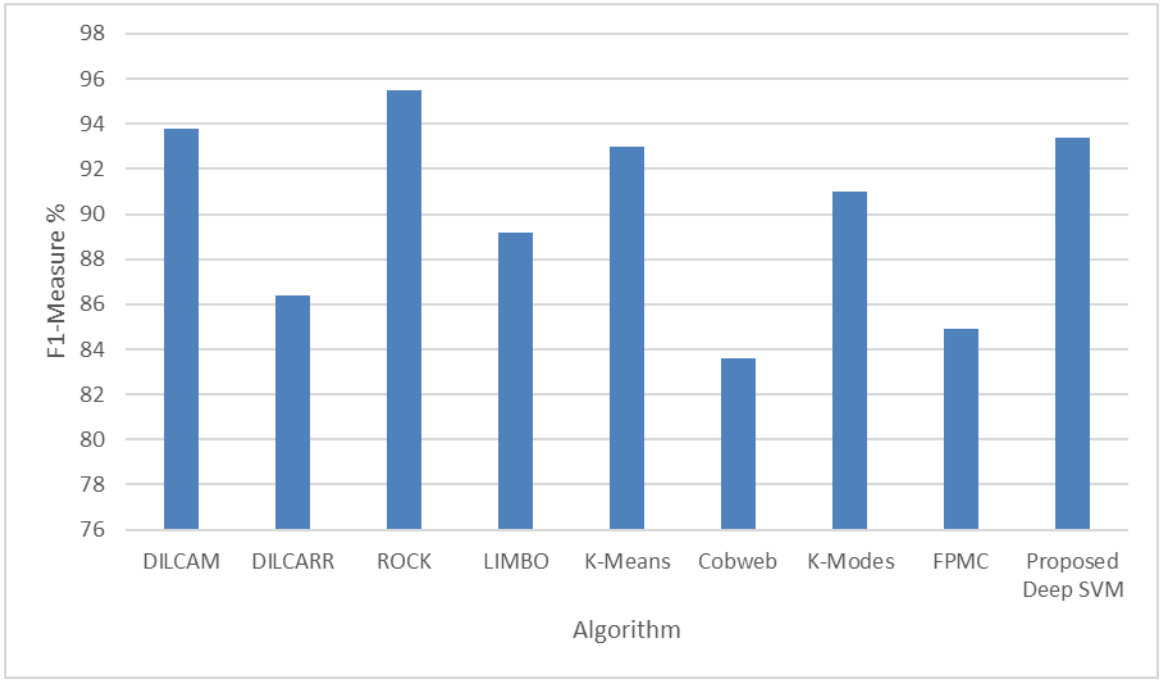

Figure 5: F1-Measure of Clustering Algorithms.

This figure 3-5 highlights that the proposed Deep SVM algorithm achieves a clustering accuracy of 80.45%, which outperforms most of the other algorithms in the comparison.

5. CONCLUSION

In conclusion, this research represents a significant stride in the realm of data mining and clustering algorithms. By exploring and enhancing established techniques such as Frequent Pattern Mining, Association Rule Analysis, Classification, and Clustering, this study has laid the groundwork for more effective data analysis. The focus on Deep SVM as a rapid and user-friendly clustering algorithm holds the promise of swift analysis of extensive, high-dimensional datasets. However, the challenges posed by real-world datasets, particularly those with substantial dimensionality, necessitate innovative solutions.This technique, tailored to balance computational efficiency and cluster quality, showcases the commitment to refining clustering methodologies. The incorporation of dynamic parameters driven by user preferences further underscores the adaptability and practicality of the proposed algorithm.By addressing issues like feature significance and dimensionality reduction, this research contributes not only to the field of data mining but also to diverse applications across industries. From business optimization through product analysis to educational interventions based on student performance, the implications are far-reaching.

REFERENCES

1. Abubaker, Mohamed, and Ashour, Wesam 2013, Efficient Data Clustering Algorithms: Improvements over Kmeans', International Journal of Intelligent Systems and Applications, 5, 3, pp. 37-49.
2. Yuvaraj, N., & Vivekanandan, P. (2013, February). An efficient SVM based tumor classification with symmetry non-negative matrix factorization using gene expression data. In 2013 International conference on information communication and embedded systems (Icices) (pp. 761-768). IEEE.
3. Asok, Aswathy, Jisha, J., T., Ashok, Sreeja and Judy, V., M. 2016, Integrated Framework Using Frequent Pattern for Clustering Numeric and Nominal Data Sets', Proceedings of International Conference on ICT for Sustainable Development, 408, pp. 523-530.
4. Suresh Ghana Dhas, C. (2020). High-performance link-based cluster ensemble approach for categorical data clustering. The Journal of Supercomputing, 76, 4556-4579.
5. Bai, Liang, Liang, Jiye, Dang, Chuangyin and Cao, Fuyuan 2013, A novel fuzzy clustering algorithm with between-cluster information for categorical data', Fuzzy Sets and Systems, 215, pp.55-73.
6. Praghash, K., Peter, G., Stonier, A. A., & Priya, R. D. (2022, December). Financial Big

Data Analysis Using Anti-tampering Blockchain-Based Deep Learning. In International Conference on Hybrid Intelligent Systems (pp. 1031-1040). Cham: Springer Nature Switzerland..

7. Borkar, G. M., Patil, L. H., Dalgade, D., & Hutke, A. (2019). A novel clustering approach and adaptive SVM classifier for intrusion detection in WSN: A data mining concept. Sustainable Computing: Informatics and Systems, 23, 120-135.
8. Sanakal, R., & Jayakumari, T. (2014). Prognosis of diabetes using data mining approach-fuzzy C means clustering and support vector machine. *International Journal of Computer Trends and Technology*, *11*(2), 94-98.
9. Kesavaraj, G., & Sukumaran, S. (2013, July). A study on classification techniques in data mining. In *2013 fourth international conference on computing, communications and networking technologies (ICCCNT)* (pp. 1-7). IEEE.
10. Yu, H., Yang, J., & Han, J. (2003, August). Classifying large data sets using SVMs with hierarchical clusters. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 306-315).
11. Dindarloo, S. R., & Siami-Irdemoosa, E. (2017). Data mining in mining engineering: results of classification and clustering of shovels failures data. *International Journal of Mining, Reclamation and Environment*, *31*(2), 105-118.
12. Yu, H., Yang, J., Han, J., & Li, X. (2005). Making SVMs scalable to large data sets using hierarchical cluster indexing. *Data Mining and Knowledge Discovery*, *11*, 295-321.