ISSN: 2633-4828 Vol. 5 No.4, December, 2023

International Journal of Applied Engineering & Technology

SCALING OF CLUSTERING ALGORITHMS USING DEEP LEARNING FOR HIGH-DIMENSION DATA

Anwiti Jain[1] and Dinesh Kumar Sahu[2]
[1]Department of Computer Science & Engg., SRK University, Bhopal, India
[2]Department of Computer Science & Engg., SRK University, Bhopal, India
[1]anwitijain@gmail.com and [2]drdineshksahu@gmail.com

ABSTRACT
Clustering algorithms are essential for discovering patterns and structures within high-dimensional data across various fields, including machine learning, data science, and artificial intelligence. However, traditional clustering methods often struggle with the challenges posed by large-scale and high-dimensional data due to their computational complexity and inefficiency. To address these limitations, this paper explores the scaling of clustering algorithms using deep learning techniques. The proposed methodology leverages deep learning models to improve the scalability and performance of clustering algorithms when dealing with high-dimensional data. Specifically, the approach involves learning low-dimensional representations of the data using deep neural networks, which can efficiently capture the underlying structure and patterns. These learned representations enable the application of conventional clustering algorithms, such as k-means, hierarchical clustering, and DBSCAN, with significantly reduced computational costs. Moreover, the paper introduces an ensemble clustering approach that combines multiple clustering results from different algorithms or multiple runs of the same algorithm. By aggregating the strengths of various clustering techniques, the ensemble method improves the robustness and accuracy of the final clustering solution. The use of deep learning also facilitates the integration of diverse clustering algorithms and parameters, leading to more stable and reliable outcomes. Extensive experiments on real-world high-dimensional datasets demonstrate the effectiveness and scalability of the proposed approach. The results show that the deep learning-based clustering algorithms can handle large-scale data more efficiently, providing accurate and meaningful clustering results with reduced computation time. This advancement has significant implications for various domains, including data analysis, image processing, and natural language processing, where high-dimensional data is prevalent. The scaling of clustering algorithms using deep learning offers a promising avenue for handling high-dimensional data more effectively. The combination of deep neural networks and ensemble clustering enhances the performance of traditional methods, paving the way for new advancements in data clustering and analysis.

Keywords: - Clustering, Ensemble, Spectral data, Deep Learning, Wavelet

INTRODUCTION
Clustering stands as a fundamental technique within data mining, aiming to organize data into cohesive groups where observations within each group, or cluster, exhibit similarity while differing from those in other clusters. Prototype-based clustering algorithms, exemplified by the widely-used K-means method, are noted for their sensitivity to initialization, which refers to the selection of initial prototypes[1,2,3]. Optimal initialization greatly influences clustering outcomes, enhancing results and reducing the number of iterations required for algorithm convergence. High-dimensional and spectral data processing presents several challenges when it comes to pattern estimation. Mining spectral data often involves utilizing various machine learning algorithms such as supervised learning, unsupervised learning, and reinforcement learning. These algorithms are frequently chosen based on their ability to handle clustering for pattern generation. However, the mentioned clustering algorithms may struggle with managing the large amounts of noise often present in spectral data. Some authors have proposed using density-based clustering approaches to handle noise and generate patterns more effectively. More recently, ensemble clustering algorithms have been introduced as a prototype approach to improve pattern generation performance. Ensemble clustering combines multiple base clustering's to create a consensus clustering those benefits from the diversity of the base models[4]. The quality of consensus clustering created by an ensemble

method is measured using external validity measures when there is prior knowledge about the underlying clustering structure of the dataset, also known as ground-truth information. In the typical clustering ensemble setup, the original data features are not utilized directly as input; instead, the quality of the consensus depends solely on the quality of the base clustering ensemble. Research indicates that a certain level of variation among the base clusters can positively influence the quality of the consensus clustering[5]. This variation allows for the integration of insights from different clusters, leading to the formation of a high-quality consensus clustering that better represents the underlying patterns in the data. High-dimensional data can be tackled using global dimension reduction techniques. A common approach involves reducing the data's dimensionality before applying traditional clustering methods. Principal Component Analysis (PCA) is the most popular technique in this category. However, PCA is limited to linear relationships between variables. Recent advancements include non-linear techniques like Kernel PCA and methods based on neural networks to address this limitation. Spectral clustering can be computationally expensive[6]. A common strategy to address this is scarifying the affinity matrix, which reduces memory usage and simplifies eigen-decomposition. However, this approach still requires calculating all entries in the original matrix. Another strategy involves sub-matrix construction[7]. The Nystrom method, for example, randomly selects a subset of data points to build a smaller affinity sub-matrix. Machine learning, particularly deep learning, has significantly advanced in recent years, providing various algorithms and models that can handle complex and noisy data and generate better patterns and insights. Deep learning models, such as convolutional neural networks (CNNs) and long short-term memory (LSTM) networks, can effectively handle noisy data. They are designed to learn robust features and patterns from data, even in the presence of noise. This ability makes them suitable for a wide range of applications, including image and audio processing, natural language processing, and time series analysis[8]. Deep learning models excel at recognizing complex patterns in data. CNNs, for instance, are particularly good at identifying spatial hierarchies in images, such as edges, shapes, and textures. LSTM networks, on the other hand, are adept at capturing temporal patterns in sequential data, such as time series or natural language[9,10]. While deep learning models are not traditionally used for clustering, they can be used in conjunction with clustering algorithms. For example, autoencoders can be employed to reduce the dimensionality of data, making it easier to apply clustering algorithms such as k-means or hierarchical clustering. Autoencoders are a type of neural network used for unsupervised learning and data compression. They consist of an encoder that transforms input data into a lower-dimensional representation (latent space) and a decoder that reconstructs the input data from the latent space. Autoencoders can be useful for reducing the spectral data and for anomaly detection. CNNs are a class of deep learning models primarily used for image and video analysis. They can automatically learn to detect important features in images, such as edges, shapes, and objects. CNNs can be adapted for other types of data with spatial or grid-like structure. LSTMs are a type of recurrent neural network (RNN) designed to handle sequential data, such as time series or natural language. LSTMs have memory cells that can retain information over long periods, allowing them to capture long-term dependencies in data. deep learning offers a variety of algorithms and models for handling complex and noisy data, reducing spectral data, and analysing series data. These models can be applied to a wide range of fields, from computer vision and natural language processing to finance and healthcare[12,13]. The combination of deep learning, specifically Long Short-Term Memory (LSTM) networks, with clustering algorithms can lead to advanced scaling techniques that improve the performance and applicability of clustering processes. Clustering is the process of grouping data points based on their similarities. Scaling in clustering refers to adjusting parameters or inputs to make the algorithm more efficient or suitable for different data sets. This can include adjusting distance metrics, feature weighting, or handling different data scales. LSTM networks, a type of recurrent neural network (RNN), are designed to handle sequential data with dependencies. When applied to clustering, LSTMs can capture temporal dependencies or patterns within the data, which can enhance the clustering process. The proposed scaling algorithm utilizes deep learning, specifically LSTM, to manage scaling factors in the clustering process. This approach can dynamically adjust parameters such as feature scaling, weights, or distance metrics based on the data and the current state of the clustering process. In this approach, LSTM can be used to propagate the ensemble process of the employed clustering algorithm. An ensemble process in machine learning involves combining

multiple models or algorithms to improve the overall performance. In this context, the LSTM network can track the progress of the clustering process and adjust scaling factors accordingly. By using LSTM to handle scaling factors, the clustering process becomes more robust to changes in data distribution and the presence of noise. This can result in more accurate and reliable clustering outcomes. Utilizing LSTM in the scaling process can lead to more efficient and effective clustering. The LSTM can capture complex patterns in the data and guide the clustering algorithm to focus on relevant features or dimensions. The objective of paper of ensemble clustering mention here.

1. The proposed ensemble clustering algorithm reduce the outlier of DBSCAN algorithm and improves the efficiency of pattern generation.
2. The proposed algorithm improves the data representation of spectral data
3. The proposed algorithm efficiently manages time complexity.

The rest of the paper organized as in section II describes related work in the area of spectral clustering; section III explores the proposed methodology of ensemble clustering; section IV explores the experimental analysis; section V provides results and discussion; and finally, section VI concludes.

II. RELATED WORK

The exploration of high-dimensional data poses significant challenges in pattern mining within the realms of data mining and machine learning. With the increase in the dimensionality of data, traditional clustering algorithms often face issues related to computational complexity, scalability, and the curse of dimensionality. To address these challenges, researchers have been increasingly turning to spectral clustering approaches. Spectral clustering techniques leverage the eigen structure of the similarity matrix derived from the data to partition it into clusters. By transforming the data into a lower-dimensional space using eigenvectors, spectral clustering can often provide more meaningful clusters in high-dimensional datasets. Several authors have explored and proposed variations of spectral clustering algorithms tailored specifically for mining high-dimensional and spectral data. Additionally, there has been a growing interest in utilizing deep learning techniques for clustering tasks. Deep learning models, particularly neural networks with multiple hidden layers, have shown promise in learning hierarchical representations of data, which can be beneficial for clustering high-dimensional datasets. Authors have proposed various deep learning architectures and optimization-based algorithms for clustering, aiming to exploit the capacity of neural networks to capture complex patterns and relationships within the data. These developments highlight a shift towards leveraging advanced computational techniques such as spectral clustering, deep learning, and optimization-based approaches to tackle the challenges posed by high-dimensional data in the context of pattern mining and clustering. Through these methods, researchers aim to improve the effectiveness, scalability, and interpretability of clustering algorithms in handling increasingly complex datasets. In [1] extend a non-convex model for RSEC and provide a solution using the majorization-minimization Augmented Lagrange Multiplier technique, inspired by recent advancements in non-convex rank minimization. [2] introduces an adaptation of this Framework to various ensemble clustering methodologies, expanding the MultiCons closed-sets-based multiple consensus clustering methodology. This extension aims to enhance the Amadeus Revenue Management Application. [3] incorporates linear ordering using ensemble clustering for symbolic data and multidimensional scaling for results display. [4] introduces an alternate four-step Optimization Technique with known Convergence. Experimental findings on various datasets demonstrate that our SMVSC technique achieves comparable or better clustering performance with significantly improved efficiency compared to both large-scale oriented methods and state-of-the-art multi-view subspace clustering methods. [5] evaluates the viability of ETH in Sachdev-Ye-Kitaev Majorana (SYK) models, which are a family of nonlocal disordered many-body interacting systems. These models can be modified from chaotic behaviour to integrability. [6] presents a novel distributed community detection methodology utilizing bagging ensemble methods' reduced complexity and variance to reveal the adjacent community hierarchy, aligning with the concept of community prediction. [7] aims to establish a data-driven approach to automate the grouping of medical terminology into clinically relevant concepts by

merging multiple data sources objectively. The proposed approach involves banding, utilizing prior knowledge from the current coding hierarchy, and combining, performing spectral clustering on an ideally weighted matrix. [8] introduces tools based on the Co-Association-Matrix produced by the Ensemble, which aids in suggesting the group of elements constituting each cluster, mitigating uncertainty associated with ensemble-clustering approaches. [9] contrasts a non-clustering hybrid model with suggested strategies by comparing various decomposition techniques with WPD. [10] conducts experiments on synthetic and real patient stratification datasets, demonstrating the effectiveness of the proposed algorithm compared to several clustering algorithms, ensemble clustering approaches, and multi-objective clustering algorithms. [11] improves the representative capacity of the learned optimal Laplacian matrix to better utilize the data's hidden high-order link information, thereby enhancing clustering performance. A powerful method with proven convergence is developed to address the optimization problem arising from this improvement. [12] confirms the proposed framework's performance with significantly greater precision using publicly accessible real-world data and on-site deployment in the Australian Energy Market Operator. [13] highlights the significance of utilizing group information and limitations information to evaluate image data and significantly improve clustering segmentation outcomes. [14] proposes scalable and parallelizable approaches suitable for addressing big problems, comparing them with K-means++ and K-means k methods using a wide range of reference and synthetic large-scale data in trials. [15] proposes two novel data clustering algorithms, U-SPEC and U-SENC, based on the word net to resolve word ambiguity and replace each word with its context-specific meaning. [16] employs a coarse-to-fine-trained topological structure to find seed points/nodes and constructs clusters using a tree-based network. [17] introduces a cluster-level surprise measure to describe the merit of a clustering reflecting degrees of both agreement and disagreement among clusters, proposing a polynomial-heuristic for choosing clustering from the ensemble that positively contribute to forming the consensus. [18] incorporates blockchain technology for data protection along with machine learning methods including various classifiers and Optimization Algorithms. [19] addresses the challenge of improving transportation options and fuel management amid rising demand for drones and air travel. [20] encourages the use of RL beyond the model-free perspective, highlighting the potential applications of RL-Algorithms such as MBRL and cooperative MARL in 6G wireless networks. [21] utilizes a similarity matrix to identify generalized eigenvectors for embedding data in low-dimensional space using unsupervised extreme learning machine (UELM) and determines optimal data partitioning using spectral clustering. [22] proposes the B++&C algorithm, a novel hierarchical clustering method that improves Moseley-Wang (MW) objectives' performance compared to traditional techniques and contemporary heuristics. [23] draws inspiration from the observation that subspaces created by the eigenvectors corresponding to the greatest eigenvalues of data matrices of distinct classes are nearly shared by various classes in the scattering transform domain. [24] suggests a feature-extraction-clustering framework utilizing ensemble clustering to label data and characterizes the heat-affected zone using the temperature profile of the heat-affected zone. [25] emphasizes the efficiency of deep neural networks in converting mappings from high-dimensional data space into a lower-dimensional feature space, leading to improved clustering through learned representations. [26] proposes an algorithm that alternates between creating clusters in the forward pass and understanding deep networks in the backward run, dividing the latent representations space using Dirichlet-Process-Mixes without prior knowledge of the number of clusters. [27] introduces a fast method for finding natural neighbours and characteristic values of data points, enabling noise detection and cutting based on critical density and reverse density. [28] combines density gain-rate with density peak clustering to create the DGPC approach, clustering samples using their characteristics extracted from a similarity graph within the context of spectral clustering. [29] demonstrates competitive performance in accuracy and optimization quality while scaling up to large problems using conventional training on fundamental classes rather than elaborate meta-learning techniques. [30] proposes a multifaceted ensemble clustering method by randomizing a scaled exponential similarity kernel to produce diverse metrics, coupled with random subspaces to generate metric-subspace pairs. [31] suggests a developing tree model to incorporate various clustering results, demonstrating the rationality of prototype examples theoretically and experimentally. [32] proposes an effective SC-Technique using a matrix completion algorithm to construct the similarity matrix quickly, along with a split

Bregman method based on the Schatten capped p-norm to recover remaining matrix elements efficiently. [33] offers a machine-learning algorithm to study and identify anomalous aero engine performance, initially modelling performance using least squares regression on an FDR dataset. [34] presents a parameter-free method (SE-ISR-PF) for automatically choosing the trade-off parameter, scalable to massive datasets through anchor-based similarity matrix construction. [35] refines individual kernel partitions and captures partition relations in a graph structure using a proxy graph, with theoretical insights provided regarding its relation to multiple kernel subspace clustering.

III. METHODOLOGY

The methodology describe involves using deep learning and spectral data decomposition to create an ensemble clustering approach with a boosting process. This method integrates deep learning algorithms and spectral decomposition to improve spatial clustering performance. Spectral data decomposition normalizes the data and transforms it into a different space where the relationships between data points can be more easily processed by the deep learning algorithm. this involves constructing a similarity matrix for the data and computing the graph Laplacian. Then, the eigenvectors (and eigenvalues) of the Laplacian matrix are computed. The top eigenvectors form a lower-dimensional representation of the data that captures important relationships. The deep learning algorithm is used to group data into several clusters. This can be achieved using different neural network architectures such as autoencoders, convolutional neural networks (CNNs), or recurrent neural networks (RNNs), depending on the nature of the data. The algorithm learns from the data representations created by the spectral decomposition. It identifies patterns and structures in the data, grouping similar data points together. The boosting ensemble clustering approach leverages the deep learning algorithm's outputs and combines them to enhance the overall clustering quality. This process is iterative, with each iteration refining the clustering results based on previous rounds' performance. The boosting process can improve spatial clustering by focusing on hard-to-cluster data points and refining cluster boundaries in each iteration. By combining deep learning with spectral decomposition, the approach can improve clustering accuracy, capturing complex data relationships. The ensemble and boosting aspects add robustness to the method by reducing dependency on a single model and mitigating the impact of outliers or noise. Adaptability: The approach can adapt to various data types and structures due to the flexibility of deep learning and the insights provided by spectral decomposition. This ensemble clustering method can be particularly useful in applications with complex, high-dimensional data and multiple clustering requirements, such as image segmentation, anomaly detection, or customer segmentation. Let me know if you would like me to provide more specific examples or delve deeper into any part of this methodology. The proposed algorithm describes in three sections. In 1st section describes data decomposition, in 2nd section describes deep learning and 3rd section finally describes algorithm.

1st section (Data Decomposition)

wavelet-based spectral data decomposition involves combining wavelet transformation and spectral decomposition techniques to analyse data. This algorithm operates in a few key steps, each involving specific mathematical computations.

Discrete Wavelet Transform (DWT): The wavelet transform decomposes the data into different frequency components at multiple scales. The most common form of wavelet transform is the Discrete Wavelet Transform (DWT).

The given data x(t), it is transformed using a wavelet function ψ(t)

$$W(a,b) = \int x(t).\psi_{a,b}x(t)\,dt$$

Here W(a,b) is the wavelet transform at scale a and translation b

$\psi_{a,b}$ is a scaled and shifted version of the mother wavelet ψ(t)

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi(\frac{t-b}{a})$$

2. create similarity matric

Similarity Matrix: Calculate the similarity (affinity) matrix S based on the wavelet-transformed data.

$$S_{ij} = P(-\frac{|xi - xj|2}{2\sigma^2})$$

Here | xi - xj| is Euclidean distance between data points xi and xj

3. calculate the graph Laplacian L form the similarity matric

L=D-S

Here D is diagonal degree matrix where $D_{ij} = \sum_j S_{ij}$

4. Perform eigen decomposition on the graph Laplacian to compute eigenvalues and eigenvectors.

$$L.v = \lambda.v$$

2nd Section (DNN)

Deep learning algorithms employed for clustering tasks by learning representations of the data that are conducive to grouping similar data points together. One common approach is to use neural networks to create embeddings of data points in a lower-dimensional space, where traditional clustering algorithms like k-means or hierarchical clustering can then be applied. The processing of deep learning algorithm describes as

The input layer of DNN is expressed by U as

$$u(t) = \sum_{i=1}^{N} Fi\delta_{uv} + \alpha_j$$

The equation the input layer 'U' of Fi is data point of spectral with wight 'δuv' and bias factor is $\alpha j.U(t)$.

The output of activation function is obtained by equation

$$\alpha = s1b1 + s2b2 + \cdots.. + snb$$

The equation describes the relation of data point a

The output of hidden layer is described as

$$V_i(t) = \begin{cases} if\ ((s1, s2, \ldots..sn) == b1, b2, \ldots\ldots\ldots..bn), retrun\ 1 \\ otherwise\ , \qquad\qquad\qquad\qquad return\ o \end{cases}$$

Here s1, s2,……..,sn is data point of cluster c1,c2,c3,……cn

The results obtained from this forwarded to weight adjustment factor as

$$w(t) = \delta_{uv} vi(t)$$

Processing of data and estimation of errors

$$e(t) = \sum_{i=1}^{N} |\,(w(t)) - \overline{w(t)}\,|$$

Here the $\overline{w(t)}$ is predicted weight and w(t) is actual weight

3d section (Ensemble Clustering)

Ensemble clustering is a method that combines multiple clustering results to improve the overall quality and robustness of clustering outcomes. This approach is also known as consensus clustering or clustering aggregation. The idea is that by leveraging the strengths of multiple clustering algorithms or multiple runs of a single algorithm with different parameters, you can obtain a better and more stable clustering solution. The process of ensemble clustering is shown in figure 1.

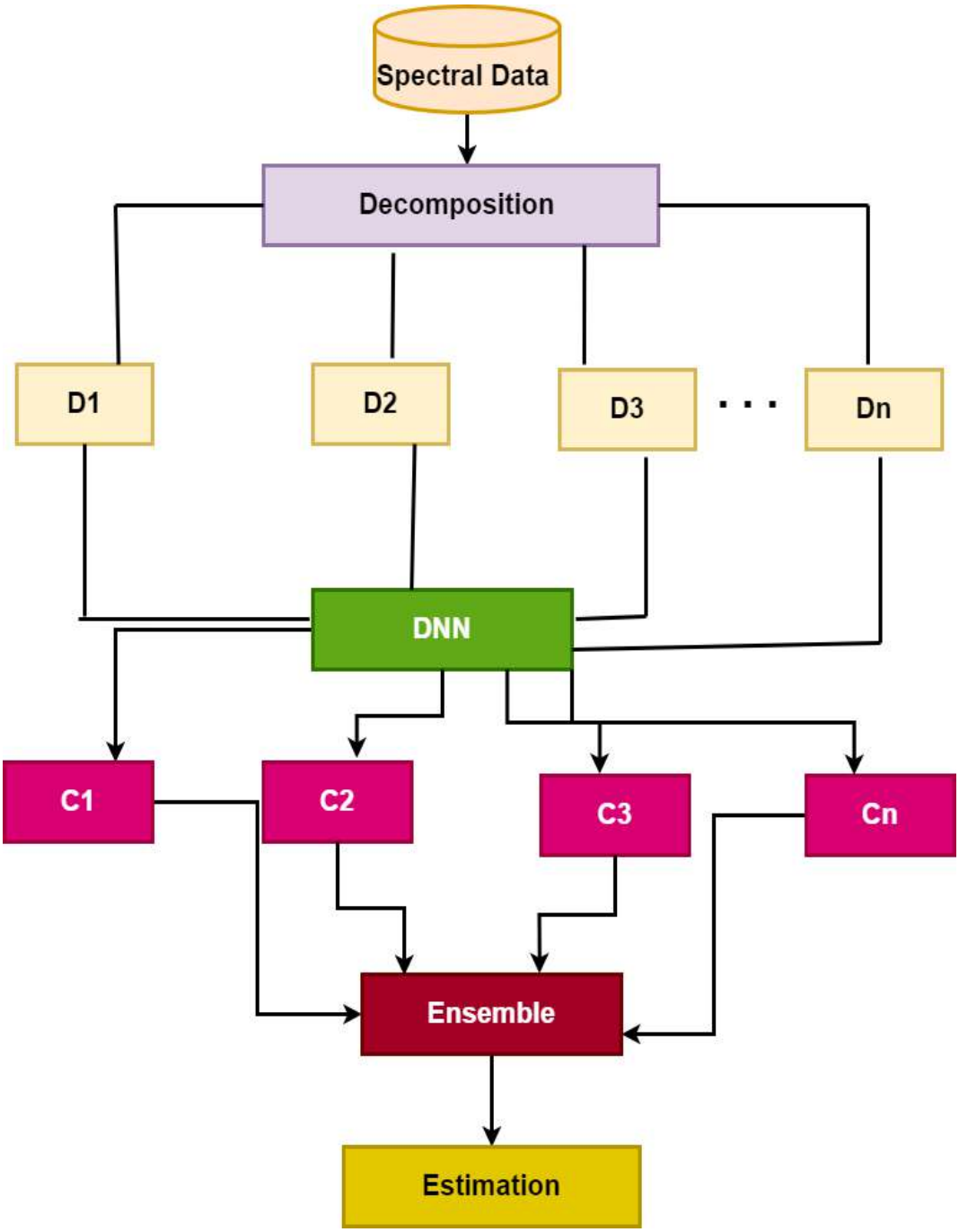

Figure 1 proposed model of ensemble clustering based on deep learning algorithm

Input Dataset $D = \{d1, d2, \ldots, d\,N\}$, DNN= $(aij)N\times K$ }

Output Consensus Clustering P with K clusters

1: $P \leftarrow \emptyset$

2: **for all** $dn \in D$ **do**

3: $Cn \leftarrow \{dn\}$ {*initialise data instance as a cluster*}

4: $P \leftarrow \{P, Cn\}$

5: $I(n) \leftarrow 1$ {*keeps track of active clusters*}

6: **end for**

7: **for** $1 \leq t \leq N - K$ **do**

8: $(s, t) \leftarrow$ arg max

$(i,j)i<j\wedge I(i)=1\wedge I(j)=1$

$A = (aij)n\times K$

9: **for** $(1 \leq u \leq N) \wedge (u \_= s \wedge u \_= t)$ **do**

10: $asu \leftarrow |Cs| \cdot asu+|Ct| \cdot atu$

$|Cs|+|Ct|$

11: $aus \leftarrow asu$

12: **end for**

13: $Cs \leftarrow Cs \cup Ct$

14: $P \leftarrow P\backslash Ct$

15: $I(t) \leftarrow 0$

16: **end for**

IV. Experimental Analysis

The proposed clustering algorithm is evaluated using MATLAB2018 software, which offers a range of functions for clustering and deep learning. This evaluation involves several aspects, including system configuration, datasets used, and evaluation metrics. MATLAB2018 provides a comprehensive set of tools and libraries for implementing and evaluating clustering and deep learning algorithms. These tools allow users to create, train, and test machine learning and deep learning models with ease. The evaluation of the proposed clustering algorithm takes place on a system with the following configuration: RAM: 16GB Operating System: Windows Processor: Intel Core i7 These specifications are sufficient for running clustering algorithms and deep learning models in MATLAB2018 efficiently. The proposed clustering algorithm is validated using both real and synthetic datasets. These datasets include: Pen Digits, USPS, Letters, MNIST: The performance of the proposed clustering algorithm is evaluated using the following metrics: The performance of the proposed algorithm is compared with existing algorithms using the same datasets. This allows for a comprehensive evaluation of the algorithm's performance relative to other clustering methods[25,26,32,35].

Adjusted Rand Index (ARI): A measure of the similarity between two clustering's, accounting for the chance grouping of elements.

Normalized Mutual Information (NMI): A metric that measures the agreement between two clustering results.

Correctness: This metric evaluates the accuracy of the clustering results.

Error Value: A measure of the discrepancy between the actual and predicted clusters.

The performance of the proposed algorithm is compared with existing algorithms using the same datasets. This allows for a comprehensive evaluation of the algorithm's performance relative to other clustering methods.

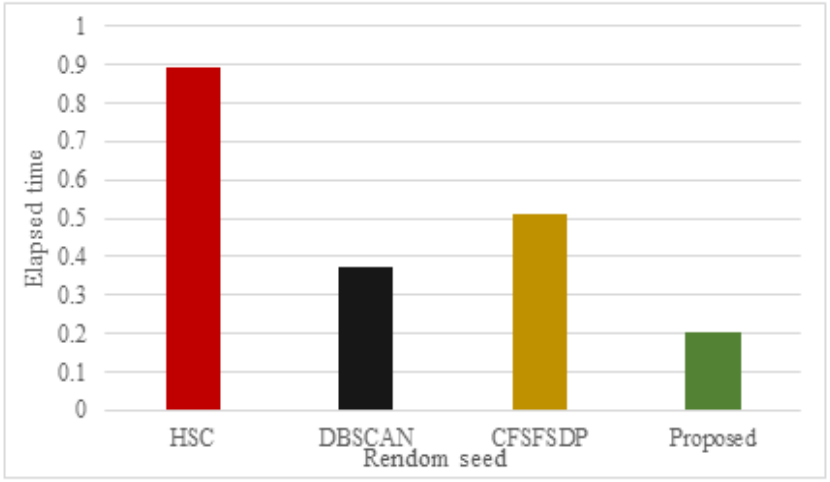

Figure 2 Performance analysis of elapsed time of proposed algorithm with HSC, DBSCAN, and CFSFDP, for PEN-DIGITS dataset.

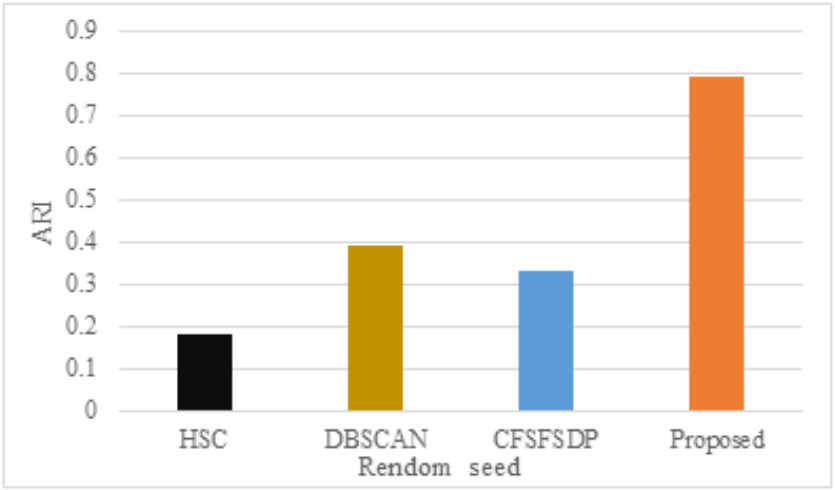

Figure 3 Performance analysis of ARI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for PEN-DIGITS dataset.

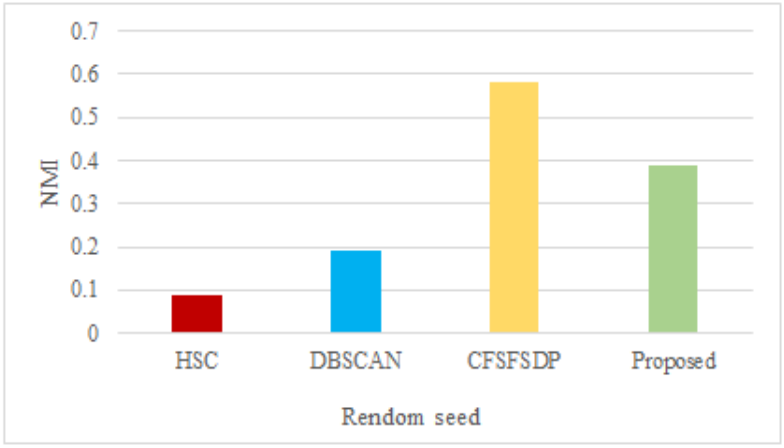

Figure 4 Performance analysis of NMI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for PEN-DIGITS dataset.

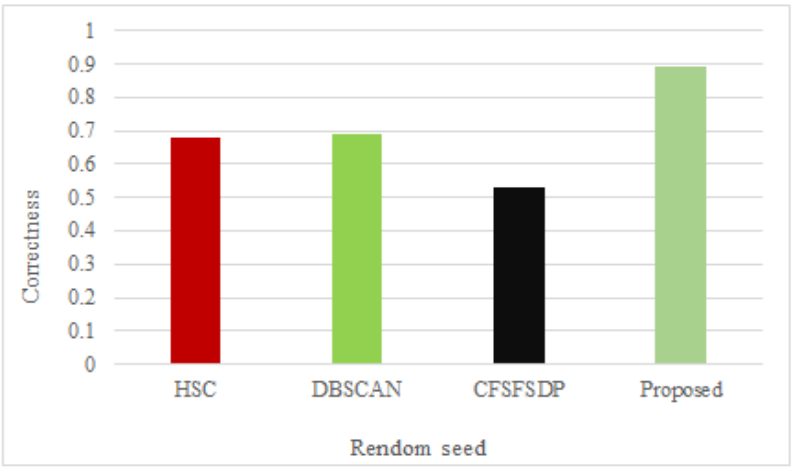

Figure 5 Performance analysis of correctness of proposed algorithm with HSC, DBSCAN, and CFSFDP, for PEN-DIGITS dataset.

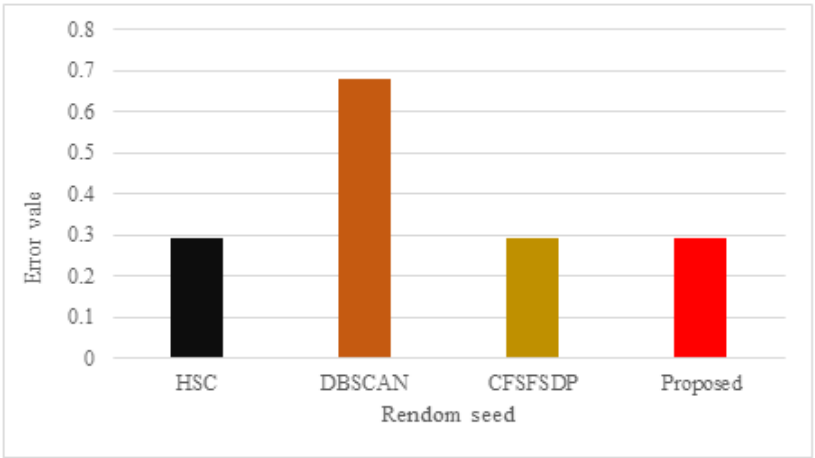

Figure 6 comparative performance of error value of proposed algorithm with HSC, DBSCAN, and CFSFDP, for PEN-DIGITS dataset.

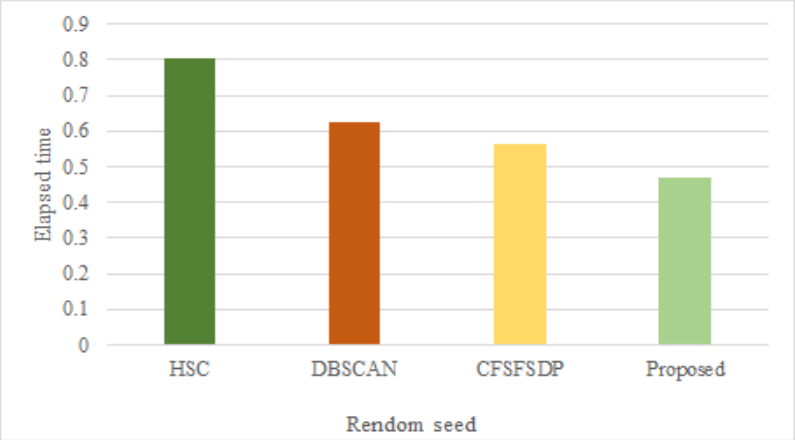

Figure 7 comparative performance of elapsed time of proposed algorithm with HSC, DBSCAN, and CFSFDP, for USPS dataset.

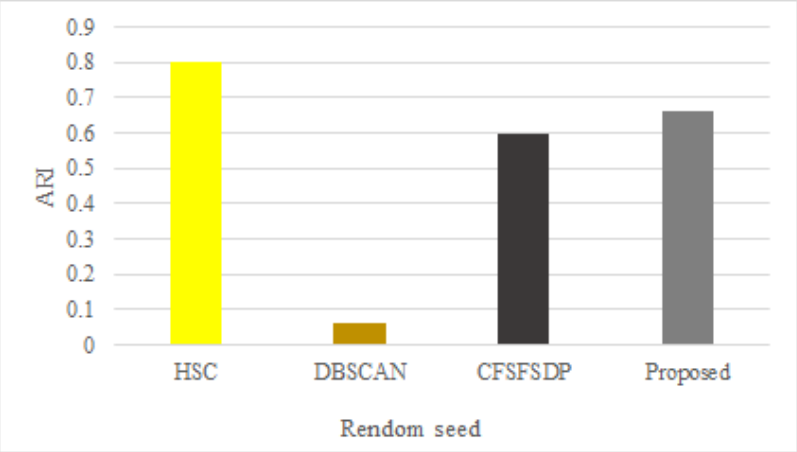

Figure 8 comparative performance of ARI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for USPS dataset.

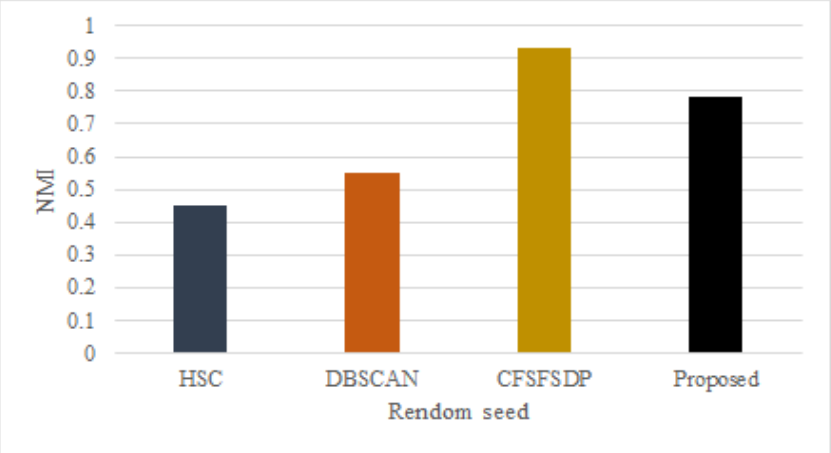

Figure 9 comparative performance of NMI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for USPS dataset.

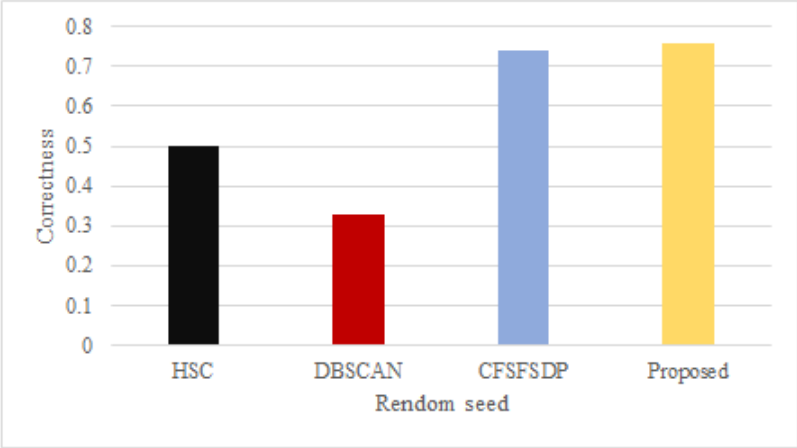

Figure 10 comparative performance of correctness of proposed algorithm with HSC, DBSCAN, and CFSFDP, for USPS dataset.

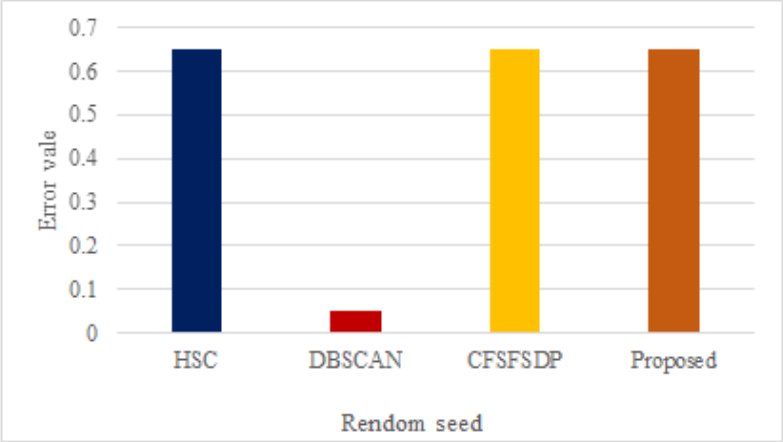

Figure 11 comparative performance of error value of proposed algorithm with HSC, DBSCAN, and CFSFDP, for USPS dataset.

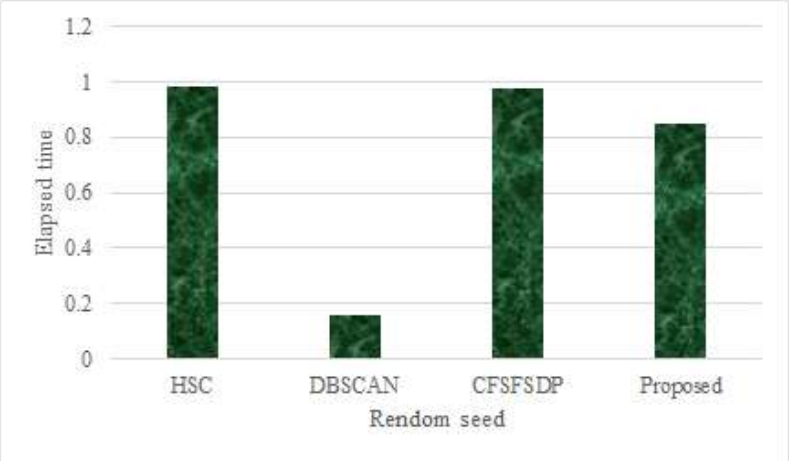

Figure 12 comparative performance of elapsed time of proposed algorithm with HSC, DBSCAN, and CFSFDP, for LETTERS dataset.

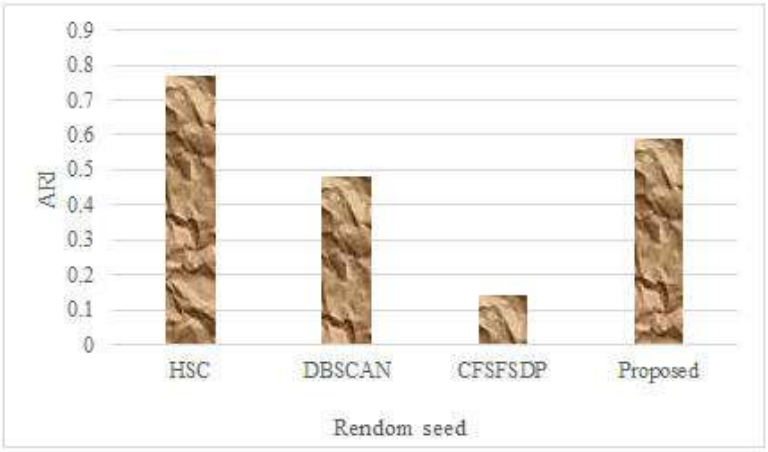

Figure 13 comparative performance of ARI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for LETTERS dataset.

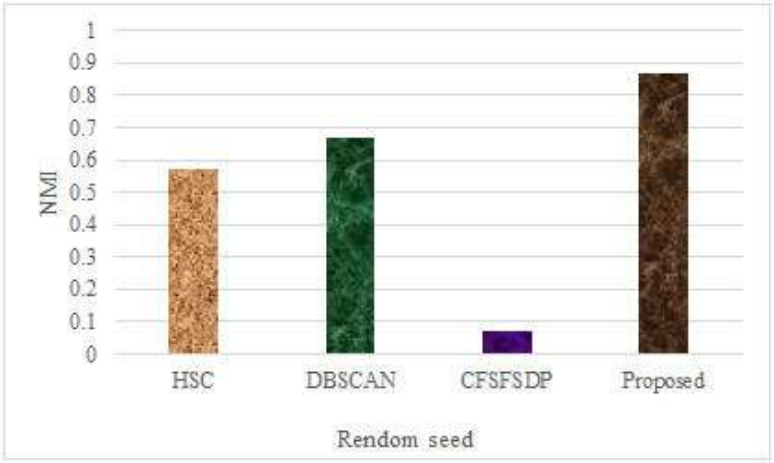

Figure 14 comparative performance of NMI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for LETTERS dataset.

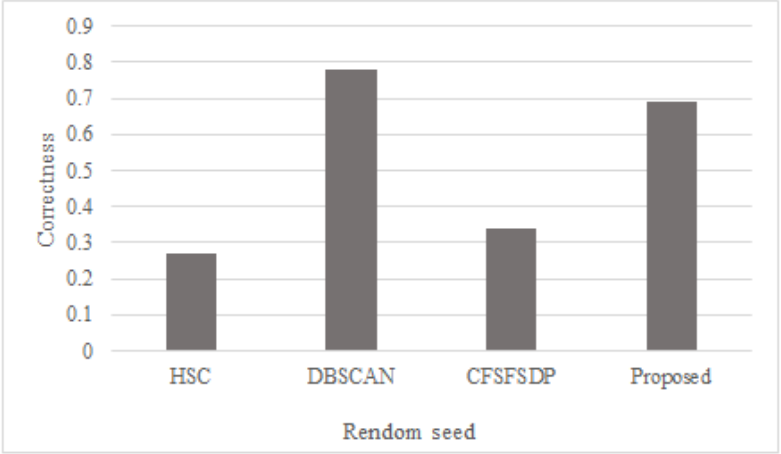

Figure 15 comparative performance of correctness of proposed algorithm with HSC, DBSCAN, and CFSFDP, for LETTERS dataset.

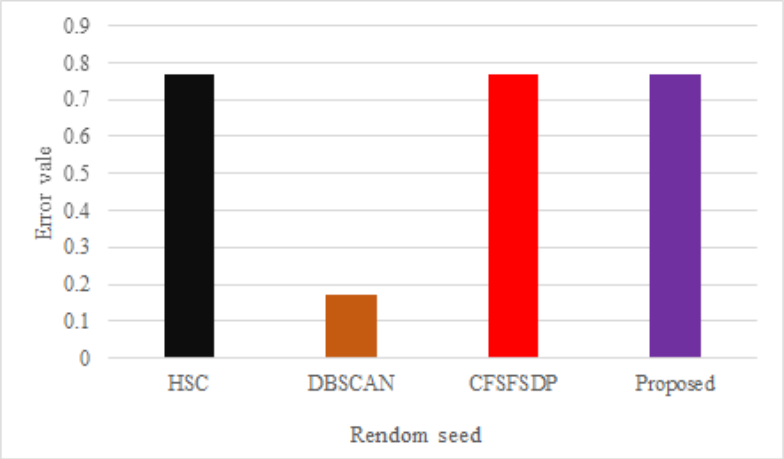

Figure 16 comparative performance of Error value of proposed algorithm with HSC, DBSCAN, and CFSFDP, for LETTERS dataset.

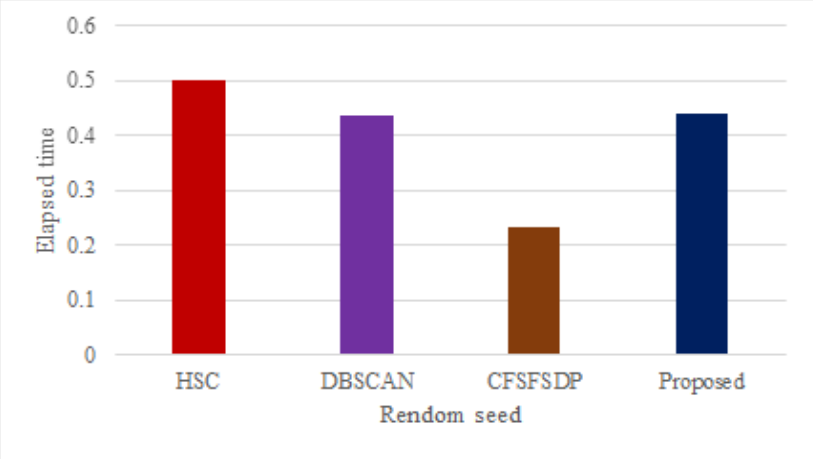

Figure 17 comparative performance of elapsed time of proposed algorithm with HSC, DBSCAN, and CFSFDP, for MNIST dataset.

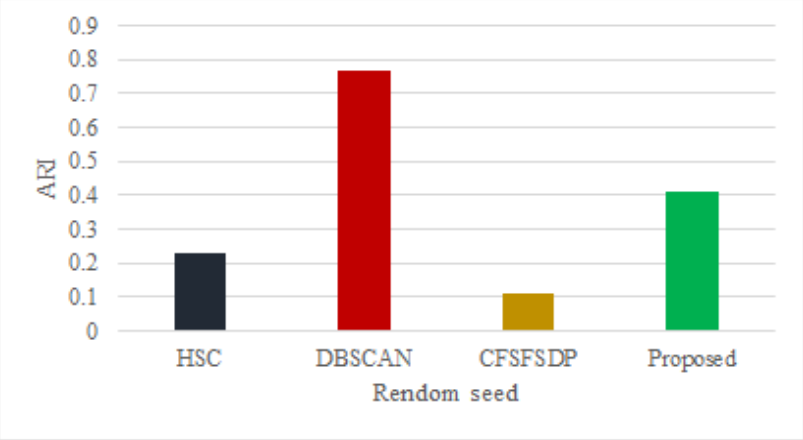

Figure 18 comparative performance of ARI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for MNIST dataset.

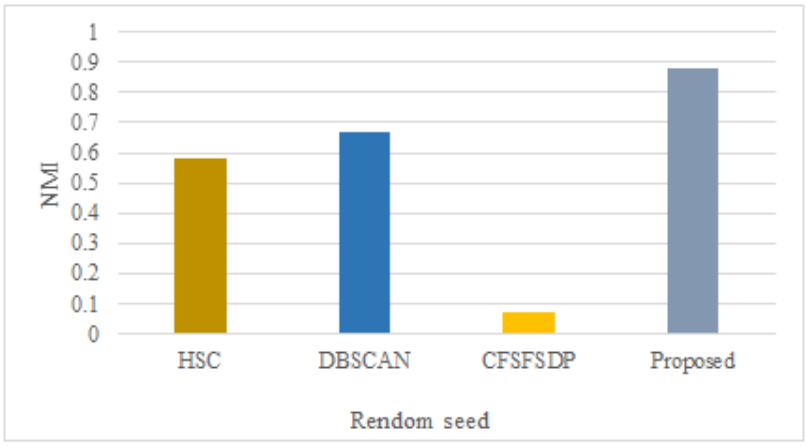

Figure 19 comparative performance of NMI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for MNIST dataset.

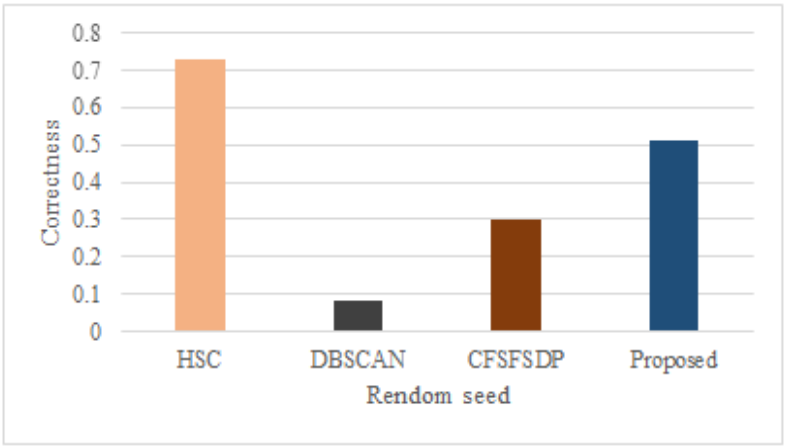

Figure 20 comparative performance of correctness of proposed algorithm with HSC, DBSCAN, and CFSFDP, for MNIST dataset.

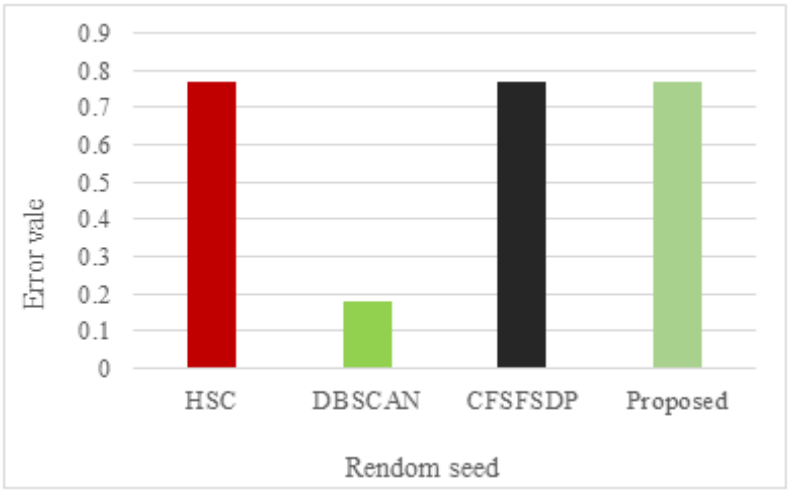

Figure 21 comparative performance of error value of proposed algorithm with HSC, DBSCAN, and CFSFDP, for MNIST dataset.

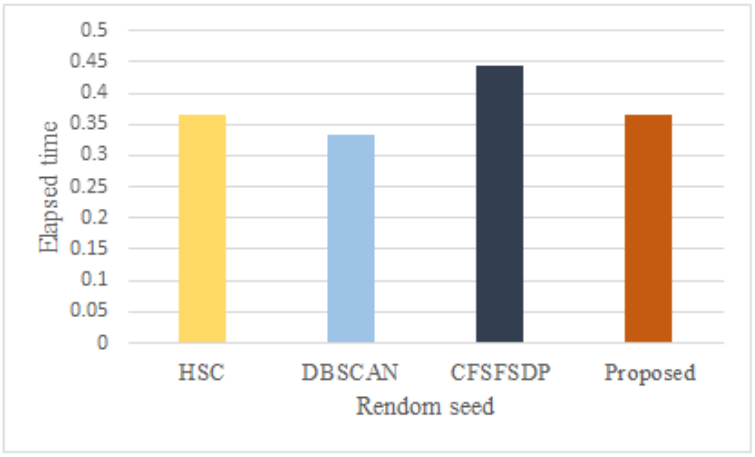

Figure 22 comparative performance of elapsed time of proposed algorithm with HSC, DBSCAN, and CFSFDP, for COVER TYPE dataset.

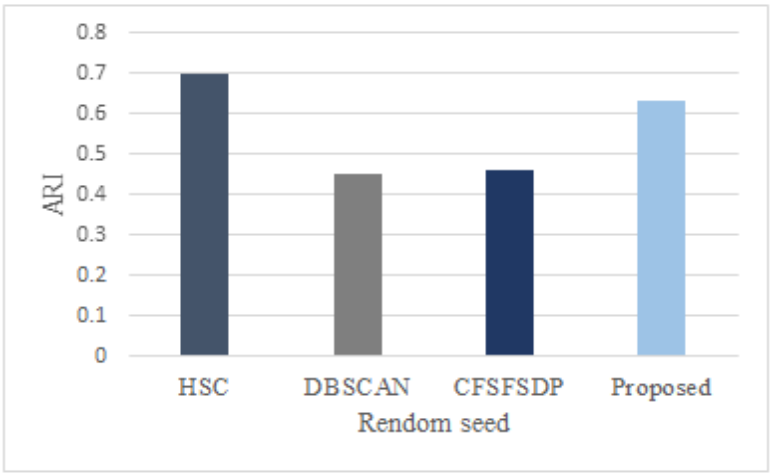

Figure 23 comparative performance of ARI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for COVER TYEP dataset.

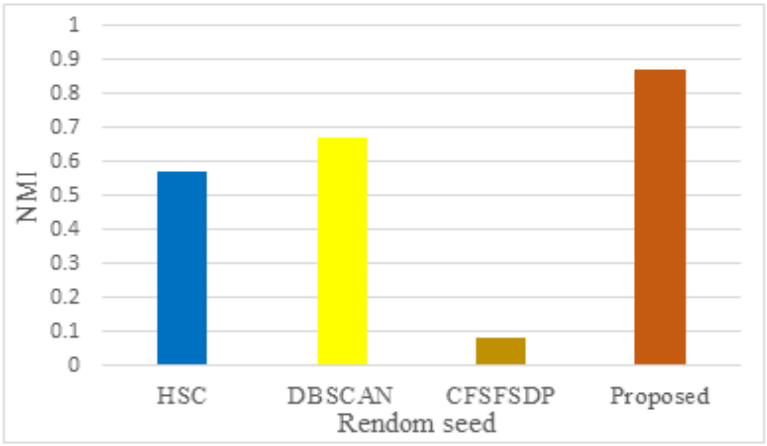

Figure 24 comparative performance of NMI of proposed algorithm with HSC, DBSCAN, and CFSFDP, for COVER TYEP dataset.

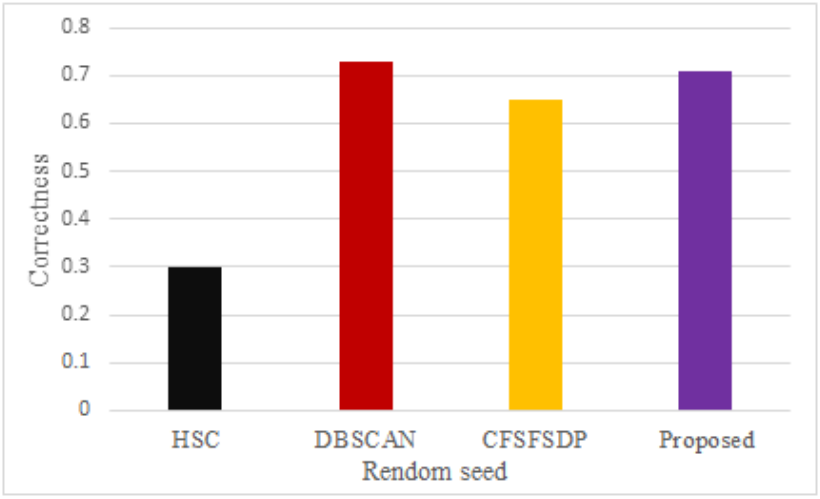

Figure 25 comparative performance of correctness of proposed algorithm with HSC, DBSCAN, and CFSFDP, for COVER TYEP dataset.

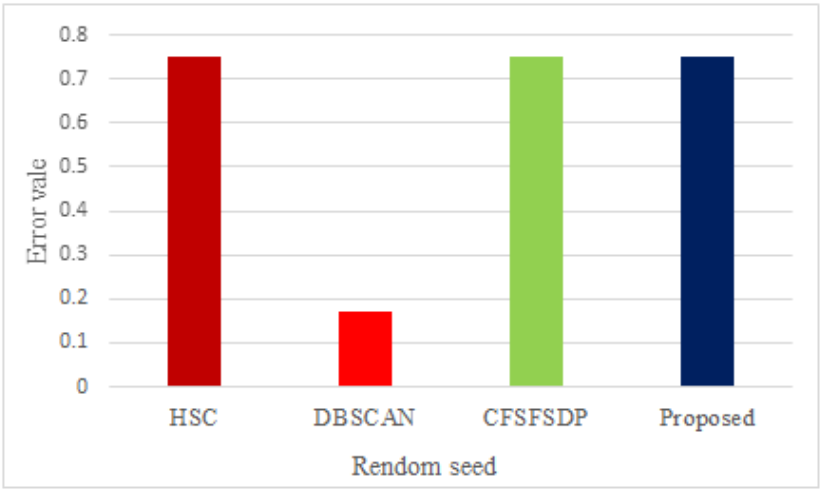

Figure 26 comparative performance of error value of proposed algorithm with HSC, DBSCAN, and CFSFDP, for COVER TYEP dataset.

V. RESULTS AND DISCUSSION

The section discusses the evaluation results of the proposed ensemble clustering algorithm for spectral data. It provides a comparison between the proposed algorithm and existing algorithms for spectral data clustering, focusing on the key performance metrics presented in Table 1. Let's explore the insights from the provided data:Table.1 Result analysis of HSC, DBSCAN, CFSFDP, and Proposed method for parameters Elapsed time, ARI, NMI, Correctness, Error value for standard datasets

Dataset	Method	Elapsed time	ARI	NMI	correctness	Error vale
pen Digits[12]	HSC	0.893620	0.180000	0.090000	0.680000	0.290000
	DBSCAN	0.374910	0.390000	0.190000	0.690000	0.680000
	CFSFSDP	0.508450	0.330000	0.580000	0.530000	0.290000

	Proposed	0.203530	0.790000	0.390000	0.890000	0.290000
USPS[]14	HSC	0.805602	0.800000	0.450000	0.500000	0.650000
	DBSCAN	0.625733	0.060000	0.550000	0.330000	0.050000
	CFSFSDP	0.562074	0.600000	0.930000	0.740000	0.650000
	Proposed	0.467539	0.660000	0.780000	0.760000	0.650000
Letters[16]	HSC	0.981867	0.770000	0.570000	0.270000	0.770000
	DBSCAN	0.158288	0.480000	0.670000	0.780000	0.170000
	CFSFSDP	0.978524	0.140000	0.070000	0.340000	0.770000
	Proposed	0.851511	0.590000	0.870000	0.690000	0.770000
MNIST[16,18]	HSC	0.498906	0.230000	0.580000	0.730000	0.770000
	DBSCAN	0.435364	0.770000	0.670000	0.080000	0.180000
	CFSFSDP	0.233966	0.110000	0.070000	0.300000	0.770000
	Proposed	0.439871	0.410000	0.880000	0.510000	0.770000
Cover type[30]	HSC	0.363883	0.700000	0.570000	0.300000	0.750000
	DBSCAN	0.332574	0.450000	0.670000	0.730000	0.170000
	CFSFSDP	0.443921	0.460000	0.080000	0.650000	0.750000
	Proposed	0.364104	0.630000	0.870000	0.710000	0.750000

The table1 provides a comparison of different clustering methods on various datasets, including Pen Digits, USPS, Letters, MNIST, and Cover type. For each dataset, the table reports the performance of four clustering methods: HSC (a specific clustering method), DBSCAN, CFSFSDP, and a proposed method. The proposed method typically achieves higher ARI and NMI scores across different datasets, suggesting better clustering performance compared to other methods. For example, on the Pen Digits dataset, the proposed method achieves an ARI of 0.79 and NMI of 0.39, which are higher than the other methods. Correctness is higher for the proposed method in most cases, indicating that it classifies data points more accurately. For instance, on the MNIST dataset, the proposed method has a correctness score of 0.51, which is higher than other methods. The error value is consistent across methods for each dataset, suggesting a potential issue with interpretation or reporting. The proposed method consistently performs well across the various datasets in terms of elapsed time, ARI, NMI, and correctness. The proposed method seems to provide a good balance between efficiency and accuracy. While the proposed method generally outperforms the other methods, there may be some trade-offs to consider, such as the error rate remaining high across all methods in some cases. The proposed method appears to be a strong contender for ensemble clustering, providing good performance metrics across different datasets. Let me know if you would like any specific analysis or further exploration of this data.

VI. CONCLUSION

In this study, we introduced a deep learning-based ensemble clustering method that selects a clustering from the ensemble based on its merit to form the consensus, ensuring that the quality of the consensus consistently improves. The merit of a clustering, an entropy measure, is calculated using a proposed cluster-level surprisal measure derived from the principle of agreement and disagreement among clusters. Empirical evidence demonstrates that our proposed approaches efficiently and effectively enhance the quality of consensus compared to established methods when considering the true number of clusters. Our findings highlight the significant impact of thoughtful ensemble selection and a clustering ensemble approach in achieving a high-quality consensus clustering. Set a novel approximation method for HSC representatives that efficiently constructs a bipartite graph linking the original data objects with a set of representatives. This approach enables the use of a transfer cut technique to achieve clustering results. Building on the CFSFSDP algorithm, we integrate multiple CFSFSDP

clusters into a unified ensemble clustering framework, referred to as the CFSFSDP algorithm. This approach leverages multiple CFSFSDP in the ensemble generation phase to create a diverse and high-quality set of base clustering's. This multiple base clustering's are then incorporated into a new bipartite graph that treats both objects and base clusters as graph nodes, which is efficiently partitioned to produce the final consensus clustering. Extensive experiments on ten large-scale datasets confirm the scalability and robustness of our algorithms.

REFERENCES

[1]. Tao, Zhiqiang, Hongfu Liu, Sheng Li, Zhengming Ding, and Yun Fu. "Robust spectral ensemble clustering via rank minimization." ACM Transactions on Knowledge Discovery from Data (TKDD) 13, no. 1 (2019): 1-25.

[2]. Yang, Tianshu, Nicolas Pasquier, and Frédéric Precioso. "Ensemble Clustering based Semi-supervised Learning for Revenue Accounting Workflow Management." In DATA, pp. 283-293. 2020.

[3]. Pełka, Marcin. "Assessment of the development of the European OECD countries with the application of linear ordering and ensemble clustering of symbolic data." Folia Oeconomica Stetinensia 19, no. 2 (2019): 117-133.

[4]. Sun, Mengjing, Pei Zhang, Siwei Wang, Sihang Zhou, Wenxuan Tu, Xinwang Liu, En Zhu, and Changjian Wang. "Scalable multi-view subspace clustering with unified anchors." In Proceedings of the 29th ACM International Conference on Multimedia, pp. 3528-3536. 2021.

[5]. Haque, Masudul, and Paul A. McClarty. "Eigenstate thermalization scaling in Majorana clusters: From chaotic to integrable Sachdev-Ye-Kitaev models." Physical Review B 100, no. 11 (2019): 115122.

[6]. Makris, Christos, Georgios Pispirigos, and Ioannis Orestis Rizos. "A distributed bagging ensemble methodology for community prediction in social networks." Information 11, no. 4 (2020): 199.

[7]. Zhang, Luwan, Yichi Zhang, Tianrun Cai, Yuri Ahuja, Zeling He, Yuk-Lam Ho, Andrew Beam et al. "Automated grouping of medical codes via multiview banded spectral clustering." Journal of biomedical informatics 100 (2019): 103322.

[8]. Fiol-González, Sonia, Cassio FP Almeida, Ariane MB Rodrigues, Simone DJ Barbosa, and Hélio Lopes. "Visual Exploration Tools for Ensemble Clustering Analysis." In VISIGRAPP (3: IVAPP), pp. 259-266. 2019.

[9]. Jiang, Feng, Jiaqi He, and Tianhai Tian. "A clustering-based ensemble approach with improved pigeon-inspired optimization and extreme learning machine for air quality prediction." Applied Soft Computing 85 (2019): 105827.

[10]. Wang, Yunhe, Xiangtao Li, Ka-Chun Wong, Yi Chang, and Shengxiang Yang. "Evolutionary multiobjective clustering algorithms with ensemble for patient stratification." IEEE Transactions on Cybernetics (2021).

[11]. Liang, Weixuan, Sihang Zhou, Jian Xiong, Xinwang Liu, Siwei Wang, En Zhu, Zhiping Cai, and Xin Xu. "Multi-view spectral clustering with high-order optimal neighborhood laplacian matrix." IEEE Transactions on Knowledge and Data Engineering (2020).

[12]. Fahiman, Fateme, Sarah M. Erfani, and Christopher Leckie. "Robust and accurate short-term load forecasting: A cluster oriented ensemble learning approach." In 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1-8. IEEE, 2019.

[13]. Xia, Kaijian, Xiaoqing Gu, and Yudong Zhang. "Oriented grouping-constrained spectral clustering for medical imaging segmentation." Multimedia Systems 26, no. 1 (2020): 27-36.

[14]. Hämäläinen, Joonas, Tommi Kärkkäinen, and Tuomo Rossi. "Scalable initialization methods for large-scale clustering." arXiv preprint arXiv:2007.11937 (2020).

[15]. Jadhav, Ms Sandhya Rangrao, and S. B. Vani. "Document Clustering On Large-Scale Data Using Ultra Scalable Spectral Clustering And Ensemble Clustering." INTERNATIONAL JOURNAL 6, no. 6 (2021).

[16]. Zhu, Shuwei, Lihong Xu, and Erik D. Goodman. "Hierarchical topology-based cluster representation for scalable evolutionary multiobjective clustering." IEEE Transactions on Cybernetics 52, no. 9 (2021): 9846-9860.

[17]. Banerjee, Arko, Arun K. Pujari, Chhabi Rani Panigrahi, Bibudhendu Pati, Suvendu Chandan Nayak, and Tien-Hsiung Weng. "A new method for weighted ensemble clustering and coupled ensemble selection." Connection Science 33, no. 3 (2021): 623-644.

[18]. Mishra, Kamta Nath, Vandana Bhattacharjee, Shashwat Saket, and Shivam P. Mishra. "Security Provisions in Smart Edge Computing Devices using Block-chain and Machine Learning Algorithms."

[19]. Singh, Ratneshwar Kumar. Blockchain Enabled Machine Learning Approach to Enhance Intelligent Transportation System. No. 6446. EasyChair, 2021.

[20]. Feriani, Amal, and Ekram Hossain. "Single and multi-agent deep reinforcement learning for AI-enabled wireless networks: A tutorial." IEEE Communications Surveys & Tutorials 23, no. 2 (2021): 1226-1252.

[21]. Homayouni, Haleh, and Eghbal G. Mansoori. "Manifold regularization ensemble clustering with many objectives using unsupervised extreme learning machines." Intelligent Data Analysis 25, no. 4 (2021): 847-862.

[22]. Naumov, Stanislav, Grigory Yaroslavtsev, and Dmitrii Avdiukhin. "Objective-based hierarchical clustering of deep embedding vectors." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 10, pp. 9055-9063. 2021.

[23]. Villar-Corrales, Angel, and Veniaming I. Morgenshtern. "Scattering transform based image clustering using projection onto orthogonal complement." In Proceedings of the 2021 Workshop on Intelligent Cross-Data Analysis and Retrieval, pp. 24-32. 2021.

[24]. Guo, Shenghan, Mengfei Chen, Amir Abolhassani, Rajeev Kalamdani, and Weihong Grace Guo. "Identifying manufacturing operational conditions by physics-based feature extraction and ensemble clustering." Journal of Manufacturing Systems 60 (2021): 162-175.

[25]. Karim, Md Rezaul, Oya Beyan, Achille Zappa, Ivan G. Costa, Dietrich Rebholz-Schuhmann, Michael Cochez, and Stefan Decker. "Deep learning-based clustering approaches for bioinformatics." Briefings in bioinformatics 22, no. 1 (2021): 393-415.

[26]. Wang, Zeya, Yang Ni, Baoyu Jing, Deqing Wang, Hao Zhang, and Eric Xing. "DNB: A joint learning framework for deep Bayesian nonparametric clustering." IEEE Transactions on Neural Networks and Learning Systems 33, no. 12 (2021): 7610-7620.

[27]. Zhang, Xiaoling, and Xiyu Liu. "Noises cutting and natural neighbors spectral clustering based on coupling P system." Processes 9, no. 3 (2021): 439.

[28]. Liu, Jiexing, and Chenggui Zhao. "Density gain-rate peaks for spectral clustering." IEEE Access 9 (2021): 46000-46010.

[29]. Ziko, Imtiaz Masud, Malik Boudiaf, Jose Dolz, Eric Granger, and Ismail Ben Ayed. "Transductive Few-Shot Learning: Clustering is All You Need?." arXiv preprint arXiv:2106.09516 (2021).

[30]. Huang, Dong, Chang-Dong Wang, Jian-Huang Lai, and Chee-Keong Kwoh. "Toward multidiversified ensemble clustering of high-dimensional data: From subspaces to metrics and beyond." IEEE Transactions on Cybernetics 52, no. 11 (2021): 12231-12244.

[31]. Li, Feijiang, Yuhua Qian, and Jieting Wang. "GoT: A growing tree model for clustering ensemble." In Proceedings of the AAAI Conference on Artificial Intelligence, vol. 35, no. 9, pp. 8349-8356. 2021.

[32]. Ma, Xu, Shengen Zhang, Karelia Pena-Pena, and Gonzalo R. Arce. "Fast spectral clustering method based on graph similarity matrix completion." Signal Processing 189 (2021): 108301.

[33]. Rajesh, M. "Aero Engine Performance Monitoring Using Least Squares Regression and Spectral Clustering." Recent Trends in Intensive Computing 39 (2021): 363.

[34]. Wang, Zhen, Xiangfeng Dai, Peican Zhu, Rong Wang, Xuelong Li, and Feiping Nie. "Fast optimization of spectral embedding and improved spectral rotation." IEEE Transactions on Knowledge and Data Engineering (2021).

[35]. Wang, Siwei, Xinwang Liu, Li Liu, Sihang Zhou, and En Zhu. "Late fusion multiple kernel clustering with proxy graph refinement." IEEE Transactions on Neural Networks and Learning Systems (2021).