Hybrid Clustering Framework for Scalable and Robust Query Analysis: Integrating Mini-Batch K-Means with DBSCAN

Hybrid Model for Complex Data Clustering

Sridevi K N[1]*, Dr. Rajanna M[2]
Assistant Professor and Research Scholar, Department of Information Science and Engineering,
Vemana Institute of Technology, Bengaluru, Karnataka, India[1]
Professor, Department of Information Science and Engineering, Vemana Institute of Technology, Bengaluru, Karnataka, India[2]

Abstract—Query clustering is a significant task in information retrieval. Research gaps still exist due to high-dimensional datasets, noise detection, and cluster interpretability. Solving these challenges will support large language models with faster and more efficient responses. This study aims to develop a hybrid clustering approach combining Mini-Batch K-means (MBK) and Density-Based Spatial Clustering of Application with Noise (DBSCAN) to cluster large-scale query datasets for information retrieval. The proposed method utilizes a preprocessing technique for data cleaning, extracts meaningful features, and scales all the features from the query dataset. The proposed hybrid clustering framework utilizes preprocessed data for clustering. The clustering algorithms MBK provide fast, scalable clustering, and DBSCAN delivers a precise, density-based refinement to efficiently process large-scale datasets while enhancing cluster boundaries to handle outliers. The proposed hybrid clustering framework effectively performs query analysis in information retrieval with a Silhouette score of 72.14 % and adjusted rand index of 78.23%. Thus, the hybrid clustering approach provides a robust and scalable solution for query analyzing tasks.

Keywords—Hybrid clustering; information retrieval; mini-batch k-means; query analysis

I. INTRODUCTION

Query analysis in a clustering framework is vital for understanding user intent and identifying behavioral patterns in information retrieval (IR) systems. Clustering is a group of objects that groups related objects into the same cluster and unrelated objects into diverse clusters. It is an analytical technique for grouping unlabeled data to extract meaningful information [1, 2]. People pursue information by asking queries on search engines. If the search engine generates unsatisfactory information, the users create another query by redeveloping the previous queries [3]. Query analysis in the clustering framework is discovering commonly asked questions and current popular topics on a search engine [4]. The clustering faces challenges in computational complexity, cluster refinement, high-data dimensionality, convergence speed, scalability, and evaluation measures [5].

Clustering is extensively utilized in pattern recognition, data mining, and query analysis, and different types of clustering algorithms have been proposed recently. The Spider Optimization Algorithm with the Sequenced User Search Pattern Query Optimization (SOA-SUSPQO) improves the efficacy of the clustering query analysis and IR [6]. The Sampling-based Density Peaks Clustering (SDPC) algorithm minimizes the distance calculations in large datasets [7]. The hierarchical clustering, k-means, and Gaussian Mixture Models (DMM) on the Domain Name System (DNS) query dataset analyze and recognize the illicit activities in the domain names [8]. The centralized clustering procedure Low-Energy Dynamic Clustering improves the energy efficacy in query-based networks that target the cluster queries in a centralized way and supports the separation of clusters that match query targets [9]. The Machine Learning (ML) approach of K-means clustering classifies the data into K groups of similar examples for information retrieval [10]. However, traditional clustering algorithms face scalability, noise detection, and cluster interpretability challenges. The proposed model solves this issue by introducing hybrid clustering algorithms MBK and DBSCAN. The motivation for the proposed work is due to raising a few research questions.

- What are the limitations of existing clustering algorithms in handling large-scale, high-dimensional, and noisy query datasets?
- How can a hybrid approach combining Mini-Batch K-means and DBSCAN improve clustering scalability, accuracy, and outlier handling?
- What are the specific performance improvements in terms of clustering quality?

The major contributions of the proposed work are summarized as follows:

- The proposed methodology develops a hybrid clustering model combining Mini-Batch K-means (MBK) for fast, scalable initial clustering and Density Based Spatial Clustering of Application with Noise (DBSCAN) for precise refinement, addressing the limitations of each algorithm.
- In the proposed methodology, we design a two-phase clustering pipeline that leverages MBK's computational efficiency to reduce processing time for large datasets.

This is followed by DBSCAN for localized, detailed analysis.

- Address a significant gap in query clustering by proposing a hybrid approach that is both efficient and effective. This will set a benchmark for future clustering techniques that deal with dynamic, noisy data in real-world environments.

The remaining section of the paper is organized as follows: Section II analyses the existing clustering algorithms, Section III describes the proposed methodology, Section IV analyses the experimental results with an ablation study, and Section V concludes the paper.

II. Literature Survey

This section reviews the performance of the existing clustering algorithms. Ates and Yaslam et al. [11] proposed a Graph-SeTES method that combines feature extraction and a decision network utilizing distance metrics and networks. The graph-based search task extraction addresses the challenges in search query logs. It improves the accuracy of short and misspelt queries, incomplete datasets, and limited labelled datasets. However, the model needed further improvements in quality embeddings. Shaik et al. [12] proposed graph-based and Machine Learning (ML) methods to improve the classification and clustering of incident ticket management systems using Resource Description Framework (RDF). The model faced limitations in scalability when utilizing a large dataset. Victor et al. [13] suggested integrated ML and PL/SQL tools to improve the database query performance. The model combined the Multi-Layer Perceptron (MLP) with k-means clustering to enhance the efficiency of the database response time.

Gong et al. [14] established a Query-Driven Clustering (QDC) protocol that leverages 5G infrastructure to improve energy efficiency and increase the network lifetime. However, the QDC algorithm required higher time complexity. To overcome this, Jia et al. [15] suggested a large-scale clustering model based on the Nystrom approximation to reduce the clustering complexity and enhance the clustering quality. Bashir et al. [16] proposed a proxy-terms-based query analysis by transforming true search queries into proxy queries that utilized the IR system, which preserves users' privacy. However, the model provided a query obfuscation only for the sensitive information contained in queries. Rehman et al. [17] suggested a Deep Learning (DL) based query response system to map farmers' queries to similar clusters. The system utilized a threshold-based clustering approach to group similar queries and enhanced the model's efficiency. The system faced difficulties with queries that had unclear meanings queries and included keywords in the dataset. Huang et al. [18] suggested a ML technique, K-means clustering, to identify the anomalous activities in the financial sector. However, the K-means clustering algorithms faced challenges in large datasets. Hartman et al. [19] developed a clustering algorithm for peptides to enhance the analysis of large spectrometry-based data that reduced the peptidomics dimensionality. The model needed further improvements in larger datasets. Zubair et al. [20] developed a model for improving the traditional K-means clustering algorithm, finding the optimal initial centroids to decrease the iterations and execution time.

Thus, the existing clustering works faced limitations in unclear and keyword queries, scalability, noise detection, large and high-dimensional datasets, and cluster interpretability. We proposed a hybrid clustering framework using the MBK and DBSCAN clustering algorithms to overcome these limitations.

III. Proposed Methodology

The framework of the proposed methodology is represented in Fig. 1.

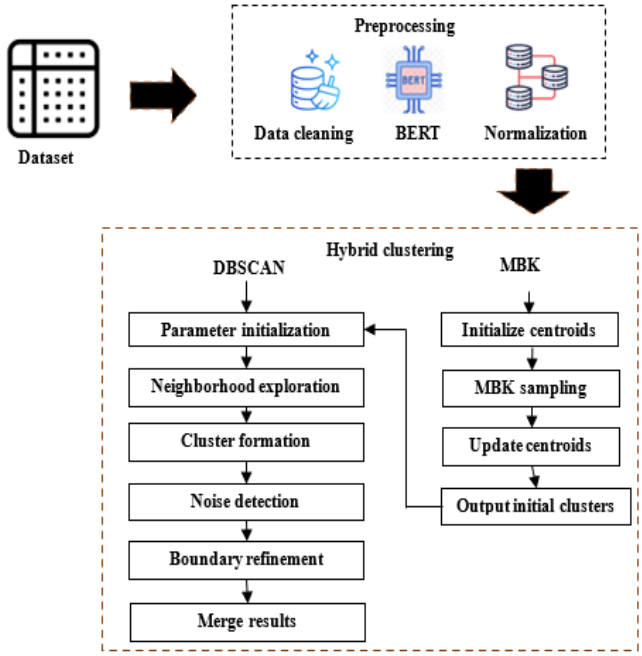

Fig. 1. Proposed hybrid clustering framework.

The proposed methodology utilizes an AOL User Session Collection 500K dataset to propose a scalable and robust query analysis framework. The query data preprocess using data cleaning, Bidirectional Encoder Representations from Transformers (BERT), and min-max normalization techniques. The data cleaning process removes the null values in the dataset, BERT extracts the meaningful features from the query data, and the min-max normalization technique scales all features in the dataset to ensure consistency across different feature dimensions. The proposed hybrid clustering algorithms MBK and DBSCAN evaluate the quality of the final clusters. The MBK clustering algorithm uses the pre-processed dataset to generate the initial clusters efficiently. The DBSCAN clustering algorithm processes each initial cluster independently to enhance cluster boundaries and detect dense regions and outliers. We utilize the hybrid clustering algorithms MBK for fast, scalable clustering and DBSCAN for precise, density-based refinement to efficiently process large datasets while enhancing cluster boundaries to handle outliers.

A. Dataset Description

The data were taken from the AOL User Session Collection 500K dataset [21]. The dataset covers 20 million web queries collected from 6,50,000 users over three months. A sequentially arranged unnamed user ID arranges the data. It provides real

query log data that is based on real users. The dataset includes four columns: AnonID, Query, QueryTime, ItemRank, and ClickURL. The AnonID contains the unnamed user ID number, the Query column contains the details of the user-issued queries, QueryTime represents the submitted time of the query, ItemRank denotes if the user clicks on a search result, the rank of the clicked item is listed, and the ClickURL listed the domain portion of the URL clicked. A query was NOT followed by the user clicking on a result item.

B. Preprocessing

Preprocessing enhances the quality of the query dataset. This research utilizes data cleaning, BERT, and min-max scaling normalization preprocessing techniques to improve the proposed query dataset.

1) Data cleaning: The data cleaning process removes the null values, such as queries with missing fields and keywords in the dataset. This process standardizes the format of the user's anonymous ID.

2) Feature engineering: BERT [22] is a language model that extracts meaningful query data features. BERT pretrains the query texts in the dataset. It breaks a sequence of tokens into characters, sentences, and words. The BERT language model pre-trained the queries in the dataset if the tokenizer did not identify the pretraining word, it split into sub words (eg: "Query"= "Que" and "ry") until the tokenizer found the sub word.

3) Min-max normalization: This research utilizes a min-max normalization [23] to scale all features in the dataset to a common range to ensure consistency across different feature dimensions. Min-max normalization performs linear transformations of the input data to generate a balance of value comparison between data after and before the process. The formulation of min-max normalization is expressed in equation (1).

$$Z_n = \frac{Z - \min(Z)}{\max(Z) - \min(Z)} \quad (1)$$

Where $min(Z)$, and $max(Z)$ denotes minimum and maximum values in the dataset, Z is the old value, and Z_n represents the new value from the normalized results.

C. Proposed Hybrid Clustering Algorithms

In this research, we propose a hybrid clustering approach that combines MBK and DBSCAN clustering algorithms that handle large-scale, high-dimensional datasets efficiently. The integration of the DBSCAN algorithm's ability to detect outliers with the MBK algorithm's computation speed to improve noise identification in large query datasets. In the first phase, the dataset is first clustered using the MBK algorithm. MBK uses mini-batches of data to update the centroids sequentially and ensure a good approximation of the clustering result. This helps speed up the process. The query dataset is first divided into a predetermined number of clusters by MBK. The overall computing load can be decreased by using this quick computation, particularly when working with big, high-dimensional query datasets. The DBSCAN algorithm is used in the second phase to refine the clusters after MBK has created the initial clusters. DBSCAN is a density-based clustering technique that can effectively manage outliers and detect clusters of different sizes and shapes by concentrating on the local density of points. The clustering process is made better overall by DBSCAN's capacity to identify and separate noise or outliers, which results in more meaningful and cohesive clusters. Additionally, it improves cluster boundary definitions that may have been ambiguous in the MBK step. Although MBK is quick, DBSCAN addresses MBK's limitations when handling noise and irregular cluster forms by fine-tuning the cluster boundaries to ensure higher accuracy in the clustering results. A hybrid framework that is accurate and computationally efficient is developed by combining the speed of MBK with the density-based refining of DBSCAN.

1) Mini-batch K-means clustering algorithm: The MBK algorithm is unsupervised learning, an improved version of the K-means clustering algorithm. It resolves clustering techniques in mixed and large datasets. This research utilizes the MBK algorithm [24] to cluster the large-scale query dataset to enhance the scalability and optimize the clustering output. The MBK requires mini-batches as input, which are random subsets of the whole dataset. The MBK has a faster computation time than the k-mean algorithm. This clustering algorithm finds the set F of cluster centers $p \epsilon R^s$ with $|F| = k$, to minimize over a set YD of examples $yd \in R^s$ the below objective function.

$$min \sum_{yd \epsilon YD} \| g(F, yd) - yd \| \quad (2)$$

In equation (2), $g(F, yd)$ returns the Euclidean distance of the adjacent cluster center $c \in F$ to yd. The problem is NP-hard, that gradient descent approach converges to the local optimum when seeded with an original set of k examples are drawn randomly from YD . The algorithm of MBK clustering is represented in pseudocode 1.

As shown in Fig. 2 MBK algorithm splits the dataset into smaller units known as mini-batches. MBK efficiently handles large datasets due to its fast computations by iteratively processing the mini-batches. In the MBK, centroids are placed as an initial marker; data points in each mini-batch are made reachable with the neighbouring centroid to locate their cluster. In the centre of the respective clusters, the centroids alter the changing distribution of data points. The final centroid generates a picture of the dataset's original clusters. Every data item fits into a certain cluster, which serves as an objective perspective for understanding, analyzing, and interpreting.

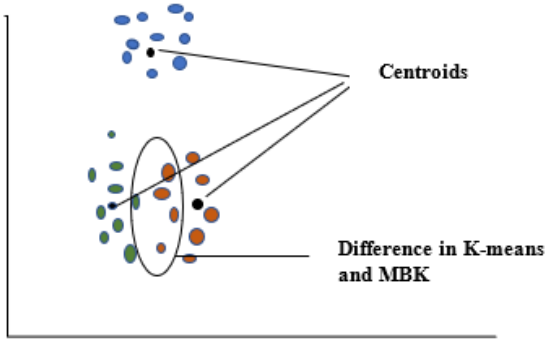

Fig. 2. Illustration diagram of the MBK clustering algorithm.

Pseudocode 1: Algorithm of MBK clustering phase

Objective: Initialize centroids for clustering
Input: $k, s \rightarrow$ mini-batch size m, iterations i, $YD \rightarrow$ data set.
Output: Set of clusters.
1: Initialize $c \in F$ with yd select randomly from YD
2: z ← 0
3: for $l = 1$ to i do
4: $S \leftarrow m$ examples select randomly from YD
5: for $yd \in S$ do
6: $d[yd] \leftarrow g(F, yd)$
7: end for
8: for $yd \in S$ do
9: $c \leftarrow d[yd]$
10: $z[c] \leftarrow z[c] + 1$
11: $\eta \leftarrow 1\, z[c]$
12: $c \leftarrow (1 - \eta)c + \eta yd$
13: end for
14: end for

2) DBSCAN Clustering algorithm: DBSCAN refers to a density-based clustering algorithm that efficiently processes the high-dimensional data and effectively distinguishes the noises in the clusters. This research utilizes the DBSCAN [25] clustering algorithm to cluster the high-density areas of target point data into clusters. It splits the data points into core, border, and noise points, as shown in Fig. 3, respectively, to the neighbourhood density points. This algorithm has neighbourhood eps and MinPts for density threshold parameters. The process of the DBSCAN clustering algorithm is denoted in pseudocode 2.

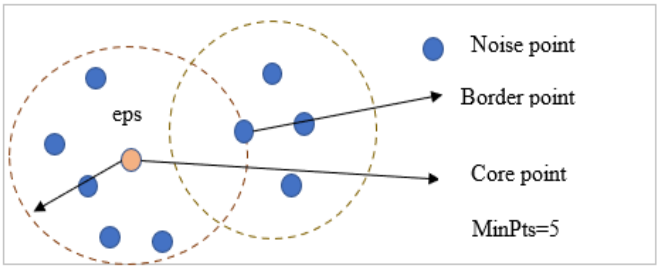

Fig. 3. Illustration diagram of DBSCAN clustering algorithm.

Pseudocode 2: Algorithm of DBSCAN clustering phase

Input: Initial MBK clustering with eps, MinPts
Output: Set of clusters
1: Randomly select a point *P*
2: Regain all points from density reachable from *P*, eps, and MinPts
3: If *P* is a core point cluster formed
4: If *P* is a core point, no points are density reachable from *P* and DBSCAN visits the next point
5: Continue the process
6: All points are processed

IV. Results

This section analyses the process and experimental results of the proposed hybrid clustering framework. The AOL User Session Collection 500K data preprocess using the data cleaning, BERT, and min-max normalization techniques. The performance of the proposed model is evaluated using the metrics of the Silhouette score, Adjusted Rand Index (ARI), and Davies-Bouldin index, and a comparative assessment analyzes the effectiveness of the proposed clustering framework. The proposed hybrid clustering framework was executed in the Python 3.10 platform on a computer with Windows 10 Pro, Intel(R) Xeon(R) CPU E5-1650 v3 @ 3.50 GHz.

A. Performance Analysis of the Proposed Model

The proposed hybrid clustering framework is evaluated using the performance metrics of the Silhouette score, ARI, and Davies-Bouldin Index. Fig. 4 illustrates the proposed hybrid clustering framework performance. The silhouette score of 72.14 % estimates the quality of clustering algorithms, the ARI of 78.23 % calculates the similarity between the two partitions of a dataset and the Davies-Bouldin Index of 86.79 % estimate the quality of the clustering models.

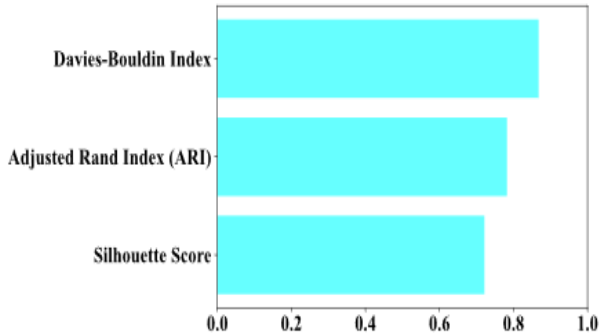

Fig. 4. Performance of the proposed hybrid clustering model.

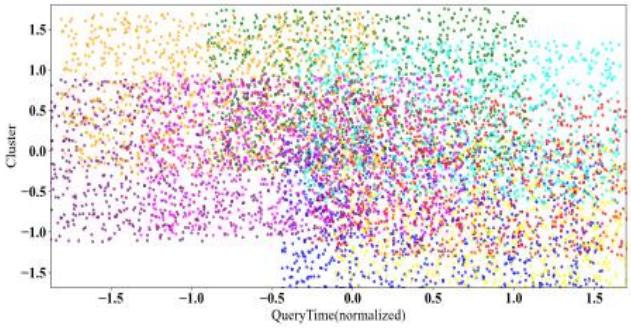

Fig. 5. Performance of the MBK initial clustering.

Fig. 5 illustrates the initial clusters generated by Mini-Batch K-means. The x-axis denotes the query time, and the y-axis represents the clusters in the query data. This represents the different clusters in distinct colors with centroids. In this clustering phase, the MBK includes various noises and outliers.

Fig. 6 demonstrates the clustering with centroids, which are represented in a group of different colours. In this graph, the x-axis signifies the query time, and the y-axis denotes the clusters in the query data. The outliers in the MBK clustering are denoted by red dots.

Fig. 7 illustrates the performance of DBSCAN clustering. It highlights DBSCAN's effect on refining clusters and detecting outliers. This graph shows how the proposed hybrid clustering algorithm efficiently reduces outliers in large-scale datasets.

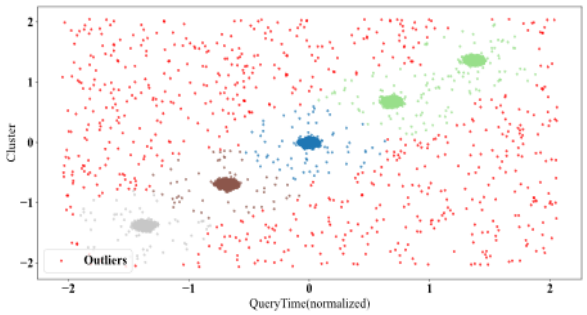

Fig. 6. Clustering with noise outliers.

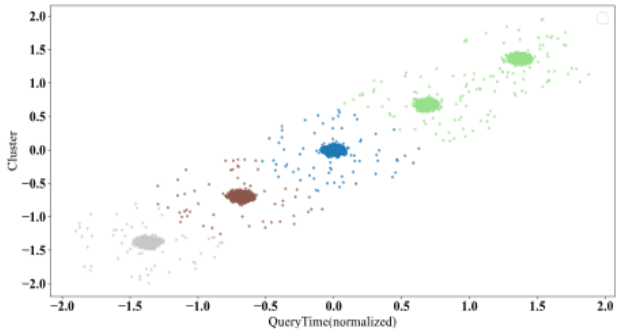

Fig. 7. Performance of the DBSCAN clustering.

Fig. 8 illustrates the performance of the queries in the dataset based on the execution time. This graph highlights the efficiency of the proposed hybrid approach. It shows that when the number of clusters increases at the same time, the execution time of the cluster also gradually increases. It is used to analyze the progression of clusters in the query data.

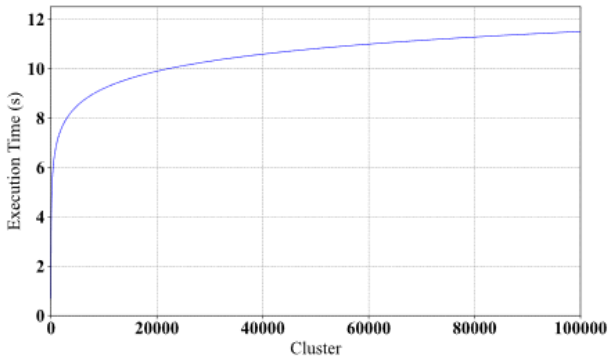

Fig. 8. Performance of the clusters based on the execution time.

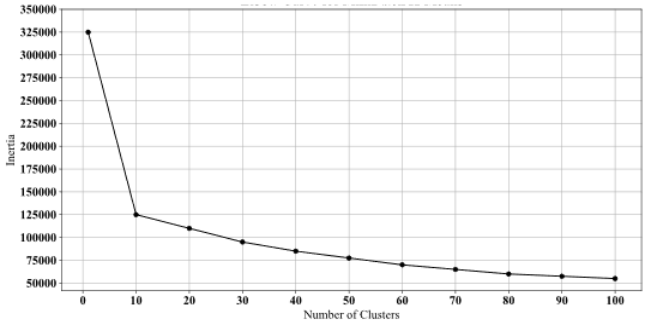

Fig. 9. Elbow curve of MBK clustering algorithm.

Fig. 9 represents the elbow curve, which shows the optimal number of clusters (k) by plotting the Within-Cluster Sum of Squares (WCSS) against diverse values of k. The visualization of the plot and the location of the elbow joint indicate the most suitable number of clusters. The MBK clustering algorithm was applied to partition the dataset into distinct clusters. The obtained clusters are denoted in the above figure.

B. Ablation Study

The ablation study evaluates the model's performance with the MBK, DBSCAN, and hybrid clustering algorithms. It highlights the importance of the proposed hybrid approach, which significantly improves the clustering quality. The MBK clustering algorithm ensures the noise detection abilities of DBSCAN and reduces the execution time. Proposed model performance with ablation study is listed in Table I.

TABLE I. PROPOSED MODEL PERFORMANCE WITH ABLATION STUDY

Methods	Silhouette score (%)	ARI (%)	Noise (%)	Execution time (s)
MBK	0.65	0.72	4.5	1.2
DBSCAN	0.72	0.75	7.2	3.5
Hybrid clustering algorithms (mini-batch &DBSCAN)	0.7214	0.7823	6.0	2.3

C. Comparative Analysis of the Proposed Model with the Existing Approaches

Table II compares the proposed hybrid clustering algorithms with the existing K-means+GMM, GMM+DBSCAN, and KisanQRS approaches.

TABLE II. COMPARATIVE ANALYSIS OF THE PROPOSED VS EXISTING CLUSTERING APPROACHES

Model	Silhouette score (%)	Davies-Bouldin Index (%)
Shaik et al., 2024, K-means+GMM [12]	0.6569	0.4614
Shaik et al., 2024, GMM+DBSCAN [12]	0.6569	0.4614
Rehman et al., 2023, KisanQRS [17]	0.6218	0.4998
Proposed	0.7214	0.8679

V. DISCUSSION

The proposed model develops a hybrid clustering approach that overcomes the limitations of traditional clustering algorithms and achieves a balance between speed, accuracy, and noise handling. As represented in Table I, the MBK clustering algorithm attains a low silhouette score of 0.65%, and ARI of 0.72% with an execution time of 1.2 s. The DBSCAN clustering algorithm alone obtains a silhouette score of 0.72%, and ARI of 0.75% with an execution time of 3.5 s. This shows the proposed hybrid clustering framework performs better with a silhouette score of 0.7214%, and ARI of 0.7823% with an execution time of 2.3 s. The MBK clustering algorithm improves the framework's scalability by processing the query dataset into smaller batches. This algorithm processes the AOL User Session Collection large-scale data effectively but has limitations in

handling noised data, and its centroids approach is complex to outliers. As shown in Table II, the existing approach achieves a Silhouette score of 0.6569 %, 0.6569 %, and 0.6218 %, while the proposed model attains a better Silhouette score of 0.7214 %, which shows that the proposed model attains a better clustering performance. The DBSCAN clustering algorithm refines the clusters and identifies the outliers effectively. This algorithm's density-based approach effectively detects the noisy points in the clusters and improves the overall clustering quality. The Silhouette score, ARI, and Davies-Bouldin index performance metrics evaluate the clustering quality. The ablation study highlights the hybrid approach significantly improves the clustering quality. It proves the noise-handling capabilities of DBSCAN while reducing the execution time through the MBK clustering algorithm. This research demonstrates the significance of hybrid clustering algorithms for scalable and robust query clustering. Training was not required for the line query when k = m because the number of clusters and data lines was equal, allowing for instantaneous data retrieval. The efficiency in the group query, k = n, where n is the number of clusters chosen, is determined by the training level: time is proportionately direct to k. After combining a few randomly chosen centroids, the data were obtained in the case of the whole query. The model's friction and fatigue ultimately serve as a representation of training consumption: friction for the k number and fatigue for the epoch amount specified in the parameters. Table III lists a few additional models that have been employed in various studies to determine the ideal k value.

TABLE III. QUALITY COMPARISON WITH OTHER SIMILAR MODELS

Ref	Model	Data	Epochs	K
[26]	Kmeans FE	50	N/A	N/A
[27]	Unsupervised Kmeans	400	11	k
[28]	Kmeans spherical	REST	N/A	k=6
Proposed	MBK and DBSCAN	100000	10	k=5

VI. CONCLUSION

In this research, we addressed the problem of clustering large-scale, high-dimensional query datasets. We proposed a hybrid clustering algorithm that provides scalable and robust noise-handling query intent detection in IR systems. The proposed methodology utilizes data cleaning, BERT, and min-max normalization techniques to extract meaningful features from the dataset. The proposed hybrid methodology begins with the MBK, which generates initial clusters by producing mini-batch sampling to handle large datasets. The DBSCAN refines the cluster boundaries and detects outliers in each cluster. The proposed hybrid algorithms overcome traditional clustering algorithms' limitations, enhancing their performance and interpretability. The experimental results demonstrate that the hybrid approach achieved superior clustering performance with a Silhouette Score of 72.14% and an ARI of 78.23%, making the model more suitable for developing LLMs. There are a number of directions for further investigation, even though the proposed approach offers notable advancements. First, advanced deep learning-based clustering methods such deep embedded clustering may improve performance even further by taking advantage of latent feature representations. Second, real-time clustering requirements in streaming data environments could be met by expanding proposed work to accommodate dynamic datasets that change over time. These possible paths provide chances to support advanced systems, improve clustering techniques, and create more effective models.

REFERENCES

[1] D. Cheng, Y. Li, S. Xia, G. Wang, J. Huang and S. Zhang, "A Fast Granular-Ball-Based Density Peaks Clustering Algorithm for Large-Scale Data," IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 12, pp. 17202-17215, 2024.

[2] Oyewole, Gbeminiyi John, and George Alex Thopil. "Data clustering: application and trends." Artificial Intelligence Review 56, no. 7 (2023): 6439-6475.

[3] Xiong, Haoyi, Jiang Bian, Yuchen Li, Xuhong Li, Mengnan Du, Shuaiqiang Wang, Dawei Yin, and Sumi Helal. "When search engine services meet large language models: visions and challenges." IEEE Transactions on Services Computing (2024).

[4] Dhanaraj, Rajesh Kumar, Vinothsaravanan Ramakrishnan, M. Poongodi, Lalitha Krishnasamy, Mounir Hamdi, Ketan Kotecha, and V. Vijayakumar. "Random forest bagging and x-means clustered antipattern detection from SQL query log for accessing secure mobile data." Wireless communications and mobile computing 2021, no. 1 (2021): 2730246.

[5] Pitafi, Shahneela, Toni Anwar, and Zubair Sharif. "A taxonomy of machine learning clustering algorithms, challenges, and future realms." Applied sciences 13, no. 6 (2023): 3529.

[6] Surya, S., and P. Sumitra. "Efficient query clustering and information retrieval using Sequenced User Search Pattern Query Optimization." Multimedia Tools and Applications (2024): 1-23.

[7] Ding, Shifei, Chao Li, Xiao Xu, Ling Ding, Jian Zhang, Lili Guo, and Tianhao Shi. "A sampling-based density peaks clustering algorithm for large-scale data." Pattern Recognition 136 (2023): 109238.

[8] Khaoula, Radi, Moughit Imane, and Moughit Mohamed. "Improving Cyber Defense with DNS Query Clustering Analysis." In 2024 11th International Conference on Wireless Networks and Mobile Communications (WINCOM), pp. 1-6. IEEE, 2024.

[9] Gong, Yadong, and Guoming Lai. "Low-energy clustering protocol for query-based wireless sensor networks." IEEE Sensors Journal 22, no. 9 (2022): 9135-9145.

[10] Purohit, Karan, Satvik Vats, Rishabh Saklani, Vinay Kukreja, Vikrant Sharma, and Satya Prakash Yadav. "Improvement in K-Means Clustering for Information Retrieval." In 2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC), pp. 1239-1245. IEEE, 2023.

[11] Ates, Nurullah, and Yusuf Yaslan. "Graph-SeTES: A graph based search task extraction using Siamese network." Information Sciences 665 (2024): 120346.

[12] Shaik, Mohammed Ali, N. Sai Anu Deep, G. Srinath Reddy, B. Srujana Reddy, M. Spandana, and B. Reethika. "Graph Based Ticket Classification and Clustering Query Recommendations through Machine Learning." Library Progress International 44, no. 3 (2024): 25828-25837.

[13] Silva-Blancas, Victor Hugo, Hugo Jiménez-Hernández, Ana Marcela Herrera-Navarro, José M. Álvarez-Alvarado, Diana Margarita Córdova-Esparza, and Juvenal Rodríguez-Reséndiz. "A Clustering and PL/SQL-Based Method for Assessing MLP-Kmeans Modeling." Computers 13, no. 6 (2024): 149.

[14] Gong, Yadong, Junbo Wang, and Guoming Lai. "Energy-efficient Query-Driven Clustering protocol for WSNs on 5G infrastructure." Energy Reports 8 (2022): 11446-11455.

[15] Jia, Hongjie, Qize Ren, Longxia Huang, Qirong Mao, Liangjun Wang, and Heping Song. "Large-scale non-negative subspace clustering based on nyström approximation." Information Sciences 638 (2023): 118981.

[16] Bashir, Shariq, Daphne Teck Ching Lai, and Owais Ahmed Malik. "Proxy-terms based query obfuscation technique for private web search." IEEE Access 10 (2022): 17845-17863.

[17] Rehman, Mohammad Zia Ur, Devraj Raghuvanshi, and Nagendra Kumar. "KisanQRS: A deep learning-based automated query-response system for

agricultural decision-making." Computers and Electronics in Agriculture 213 (2023): 108180.

[18] Huang, Zengyi, Haotian Zheng, Chen Li, and Chang Che. "Application of machine learning-based k-means clustering for financial fraud detection." Academic Journal of Science and Technology 10, no. 1 (2024): 33-39.

[19] Hartman, Erik, Fredrik Forsberg, Sven Kjellström, Jitka Petrlova, Congyu Luo, Aaron Scott, Manoj Puthia, Johan Malmström, and Artur Schmidtchen. "Peptide clustering enhances large-scale analyses and reveals proteolytic signatures in mass spectrometry data." Nature Communications 15, no. 1 (2024): 7128.

[20] Zubair, Md, MD Asif Iqbal, Avijeet Shil, M. J. M. Chowdhury, Mohammad Ali Moni, and Iqbal H. Sarker. "An improved K-means clustering algorithm towards an efficient data-driven modeling." Annals of Data Science 11, no. 5 (2024): 1525-1544.

[21] https://www.kaggle.com/datasets/dineshydv/aol-user-session-collection-500k

[22] Panoutsopoulos, Hercules, Borja Espejo-Garcia, Stephan Raaijmakers, Xu Wang, Spyros Fountas, and Christopher Brewster. "Investigating the effect of different fine-tuning configuration scenarios on agricultural term extraction using BERT." Computers and Electronics in Agriculture 225 (2024): 109268.

[23] Jlassi, Oussama, and Philippe C. Dixon. "The effect of time normalization and biomechanical signal processing techniques of ground reaction force curves on deep-learning model performance." Journal of Biomechanics 168 (2024): 112116.

[24] Purba, Andrew Castello, and Teny Handhayani. "Comparison of K-Means, Affinity Clustering, and Mini Batch K-Means Algorithms for Market Segmentation Analysis." Komputa: Jurnal Ilmiah Komputer dan Informatika 13, no. 1 (2024): 54-63.

[25] Han, Jianfeng, Xuefei Guo, Runcheng Jiao, Yun Nan, Honglei Yang, Xuan Ni, Danning Zhao et al. "An automatic method for delimiting deformation area in insar based on hnsw-dbscan clustering algorithm." Remote Sensing 15, no. 17 (2023): 4287.

[26] Benaimeche, M.A., Yvonnet, J., Bary, B. and He, Q.C., 2022. A k-means clustering machine learning-based multiscale method for anelastic heterogeneous structures with internal variables. International Journal for Numerical Methods in Engineering, 123(9), pp.2012-2041.

[27] Sinaga, K.P. and Yang, M.S., 2020. Unsupervised K-means clustering algorithm. IEEE access, 8, pp.80716-80727.

[28] George, S., Seles, J.K.S., Brindha, D., Jebaseeli, T.J. and Vemulapalli, L., 2023. Geopositional Data Analysis Using Clustering Techniques to Assist Occupants in a Specific City. Engineering Proceedings, 59(1), p.8.