An Approach For Mining Large Dataset Using Clustering Algorithm

***S Archana*[1], Dr.Neeraj Sharma [2], Dr. Pradosh chandra patnaik[3]**

[1]*Research Scholar, Dept. of Computer Science and Engineering*

Sri Satya Sai University of Technology and Medical Sciences,

Sehore Bhopal-Indore Road, Madhya Pradesh, India.

[2]*Research Guide, Dept. of Computer Science and Engineering*

Sri Satya Sai University of Technology and Medical Sciences,

Sehore Bhopal-Indore Road, Madhya Pradesh, India.

[3]*Research Co-Guide, Professor & Principal . Dept. of Computer Science and Engineering*

Aurora's PG College (MCA) Hyderabad

ABSTRACT

Clustering is an unsupervised machine learning technique for discovering and grouping related data points in big datasets without regard for the end result. Clustering is advantageous in data mining because it enables the discovery of groups and the identification of relevant distributions within the underlying data. Traditionally used clustering techniques either prefer spherical clusters with similar sizes or are extremely brittle in the presence of outliers. Utilizes a combination of random sampling and partitioning to manage huge databases. After partitioning a random sample of the data set, each partition is somewhat clustered. After that, the partial clusters are clustered again to get the desired clusters. Numerous parallel methods based on the MapReduce architecture have been presented recently to address the scalability issue caused by increasing data sizes. When huge data is clustered in parallel using the KMeans algorithm, it is read repeatedly during each iterative step, considerably increasing both I/O and network costs. We offer a new Collection-based KMeans clustering technique, dubbed CBKMeans, in this study that effectively reduces data size while improving clustering accuracy through representative

verification. Our experimental results demonstrate that CBKMeans are more efficient, scalable, and accurate than k-means.

Keywords : Data mining, knowledge discovery, clustering algorithms, sampling

INTRODUCTION

Constant advancements in science and technology have made data collecting and storage easier and more affordable than ever before. This resulted in the emergence of massive datasets in science, government, and industry, which must be processed or sorted in order to extract usable information. Consider the results returned by a search engine for a given query. The user must sift through the lengthy lists in order to locate the required solution. However, if there are millions of web sites offered as solutions to a particular query, this task might become quite challenging for the user. Thus Clustering algorithms can be extremely beneficial for gathering closely similar solutions to a given query and displaying the results in the form of clusters, allowing for the avoidance of irrelevant documents even without looking at them. The primary goal of clustering any set of data is to discover inherent structure in the data and interpret it as a collection of groups, with the data objects within each cluster exhibiting a high degree of similarity, referred to as intra-cluster similarity, while the similarity between different clusters is minimised.

Clustering is used in a variety of contexts, including the following:

- News articles: Sorting daily news stories into categories such as sports, highlights, business, and health, among others.
- Classification of online documents (WWW): Search engine results can be clustered based on their degree of similarity to the provided query.
- Exploring the market: Given a huge database including the past purchase records of each individual consumer, identifying groupings of customers who exhibit similar behaviour.
- Research projects: Collecting enormous amounts of data from sensors on a daily basis is pointless unless specific conclusions are drawn. Identifying and identifying necessary relationships in acquired data could lead to useful conclusions.
- Earthquake studies: Clustering observed earthquake epicentres to identify risky zones.

The standard clustering approaches (e.g., K-means, hierarchical clustering) are incapable of dealing with this growing volume of data. The reason for this is primarily due to the limitation of retaining all data in main memory or the algorithms' temporal complexity. As a result, they are

***e*ISSN** 1303-5150

impractical for processing these increasingly massive datasets. This indicates that the need for scalable clustering methods is a significant issue, necessitating the development of some novel methodologies. There are numerous strategies for scaling clustering algorithms, some of which are inspired by successful supervised machine learning techniques, while others are tailored specifically for this unsupervised task. For example, some of these techniques employ a variety of sampling procedures in order to retain only a part of the data in memory. Others are based on partitioning the entire dataset into many independent batches for separate processing and then merging the results to create a consensus model. Certain approaches presumptively presume that data is a continuous stream that must be processed in real time or in successive batches. Additionally, these strategies are incorporated in a variety of ways depending on the clustering model utilised. This breadth of techniques necessitates defining their qualities and organising them coherently.

Figure 1 Clustering

Cluster analysis is a commonly used core job in exploratory data analysis, with applications spanning from image processing to speech processing, information retrieval, and Web applications. As a fundamental approach, clustering is a technique that seeks to organise datasets (objects) into clusters in such a way that comparable things are aggregated together in the same

cluster, while dissimilar objects belong to separate clusters. Additionally, from an optimization standpoint, the primary objective of clustering is to maximise both (internal) homogeneity within a cluster and (external) heterogeneity across distinct clusters. In contrast to classification, clustering does not learn a classification model using a subset of data objects with known class labels. In machine learning parlance, clustering is a type of unsupervised job that determines the similarity between items without knowledge of their correct distribution. Clustering is regarded as one of the most difficult assignments because to its unsupervised nature. The various algorithms created over the years by academics result in distinct clusters of data; even when the same algorithm is used, the choice of different parameters or the presentation order of data items can have a significant effect on the final clustering partitions.

When it comes to lowering the amount of the dataset, there are two dimensions to consider. The first is the total number of occurrences. When it is evident that a smaller selection of the data contains the same information as the entire dataset, sampling techniques can be used to address this issue. This is not always the case, and sometimes the mining method is designed to identify certain groupings of cases with a low frequency but a high value. This data may be discarded throughout the sampling phase, rendering the method ineffective. In other applications, the data is in the form of a stream; this complicates the sampling procedure or increases the danger of missing critical information in the data if its distribution varies over time. The quantity of attributes in a dataset can also be handled using dimensionality reduction techniques. There are various fields of research that are concerned with the transformation of a dataset from its original representation to one with a reduced collection of features. The objective is to create a new dataset that retains some of the original structure of the data, such that its analysis produces the same or similar patterns as the original data. In general, there are two approaches to attribute reduction: feature selection and feature extraction.

REVIEW OF THE LITERATURE

Bozdemir et al., (2021) On education datasets, a parallel clustering technique based on message passing interface (MPI) called M-K-Means is applied. M-K-Means is composed of MPI and Sequential K-Means. Additionally, the K-Mean and Message Passing interfaces are employed concurrently in the same experiment, which is likewise based on random centroids selection and divides a dataset into "p" sub-datasets, where "p" is the number of nodes connected to the main computer. The approach is validated on a DNA dataset and the Simple K-Mean algorithm is parallelized with randomly chosen initial centroids. When the data set is tiny, the classic clustering technique performs well and produces satisfactory results. However, it struggles with

huge datasets due to a variety of reasons, including data volume, data dimensionality, computational power, and memory. Numerous strategies have been developed to increase the efficiency and accuracy of clustering results in a large amount of data environment.

Wang et al., (2021) conducted a similar analysis and discovered that a threshold value is used to create this new list. When the new list's items meet the threshold, the new list's values are returned as initial centroids. The Parallel K-Mean clustering algorithm is implemented using the Para Means programme. They parallelize the Simple K-Mean clustering technique for common laboratory application. Para Means is a client-server application that is simple to manage. Cluster analysis has been extensively studied as a subfield of statistics for many years. Numerous statistical analysis software packages or systems now include analysis tools. Multiple dimensions or attributes may exist in a database. Numerous clustering techniques are efficient in terms of processing. Generally, two to three dimensions are involved in low-dimensional data. In general, the quality of clustering can be accurately assessed only in three-dimensional scenarios. Clustering data items in a high-dimensional space is extremely difficult, much more so when the data is severely skewed and sparse. Capacity for noisy data processing: The majority of data in practical applications include outliers, such as missing, unknown, or inaccurate data. Certain clustering methods are sensitive to this type of data, resulting in low-quality clustering results.

Karwan Qader et al. (2017) investigated three distinct data mining techniques, including K-Means, Fuzzy C Means, and EM, as well as a 44-class strategy for network fault classification. The established approach benefited in identifying anomalous behaviours in communication networks and provided a means of real-time fault classification and management. Then, datasets from networks with high and low traffic volumes were collected, and a prototype was created for performing network traffic fault classification under the provided conditions. k-Means and EM algorithms significantly reduced the processing overhead compared to other standard techniques. However, the time complexity remained over the acceptable level. By optimising the findings of network and system measurements, the similarity measure addressed the shortcomings of existing techniques. Additionally, the similarity measure aided in traffic clustering. However, it was unable to forecast traffic flows.

Yu Wang et al. (2016) devised a limited clustering approach to improve traffic clustering accuracy. The decisions were constructed using a constrained clustering approach in conjunction with background traffic statistics. Additionally, a set of similar constraints was included in the constrained clustering technique. Then, to maximise the evaluation of algorithm parameters, we employed Gaussian mixture density and an approximate approach called the SBCK algorithm on

***e*ISSN** 1303-5150

the observed data with limits. Additionally, the effect of unsupervised characteristics on clustering was recognised using a fundamental binning method. As a result, while the limited clustering technique improves the CA, it also raises the temporal complexity.

METHODOLOGY OF RESEARCH

This research paper implements and examines Collection-based K-Mean (CBKM) algorithms, which are capable of finding clusters more quickly than normal K-means clustering methods. When the KMeans technique is used, the number of iterations required to converge the set of centres increases exponentially. Restarting MapReduce tasks, re-inputting the original data set, and shuffling intermediate results will occur repeatedly when large-scale data is clustered in parallel using the KMeans algorithm via MapReduce, considerably increasing both I/O and network expenses. As a result, a new collecting procedure is developed in order to minimise the amount of the input data. Simultaneously, the grid division approach will be considered a supplement to help reduce the time required to do a range query. The collecting algorithm will be enhanced, and representative verification will be considered. As the amount of data increases, the algorithm's time cost will become tremendous. The collection technique seeks to obtain a significantly smaller subset of points that accurately reflects all of the points and reduces the data size. Once we have the sample set, we will run KMeans on the sample points with representative verification. The collection algorithm can help reduce the size of the data and accelerate the clustering algorithm.

CBKM Algorithm

INPUT : Array {A1,A2…..An}

A = data points

N = number of required clusteres

OUTPUT : a set of clusters

STEPS :

1) Randomly select A data points from dataset B as initial centers.
2) Calculate the distance between each data point Ai (1<=i<=n) and all N clusters Cj(1<=j<=N) and assign data set object Ai to the nearest cluster
3) For each cluster j (1<=j<=N), recalculate the cluster center by taking arithmetic mean of each cluster

4) Repeat until no change in the center of cluster

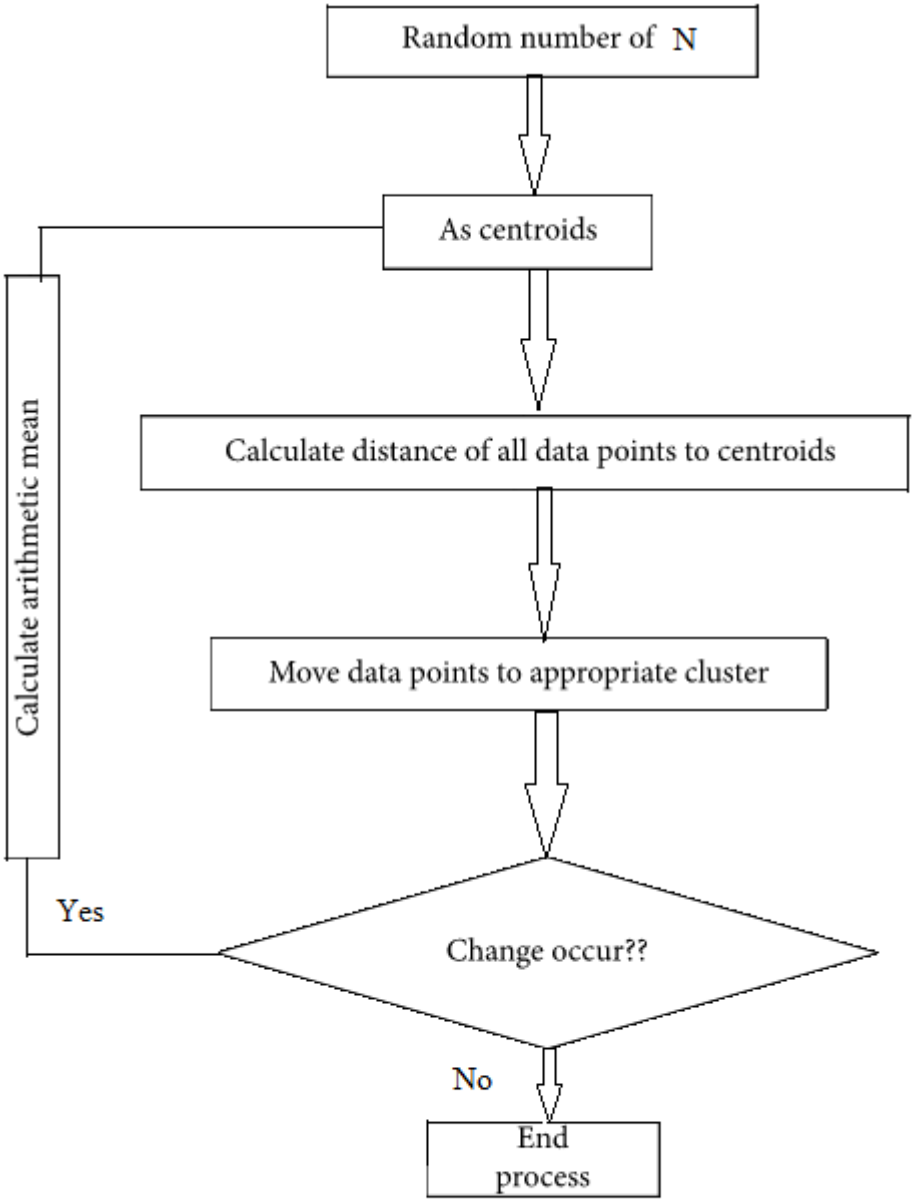

Figure 2 Flowchart of CBKM

The clustering method begins with each input point as a separate cluster and combines the closest pair of clusters at each subsequent phase. To compute the distance between two clusters, c representative points are maintained for each cluster. These are found by selecting c evenly distributed points inside the cluster and then downsizing them by a fraction a' toward the cluster

mean. The distance between two clusters is then calculated as the distance between the two clusters' nearest pair of representative points - one from each cluster.

Thus, only the cluster's representative points are used to determine its distance from other clusters. The c representative points make an attempt to capture the cluster's physical structure and geometry. Additionally, via factor cy, reducing the scattered points toward the mean eliminates surface anomalies and mitigates the consequences of outliers. This is because outliers are often located more away from the cluster centre, and as a result, shrinking causes outliers to migrate closer to the centre, while remaining representative points undergo modest alterations. The bigger movements of the outliers would therefore diminish their potential to integrate the incorrect clusters. Sampling is based on the assumption that just a random subset or subsets of the data are required to obtain a model for the data and that the entire dataset is available from the start. The random subsets may be disjoint or not. If multiple samples of data are processed, the subsequent samples are integrated into the present model. This is typically accomplished through the use of an algorithm capable of processing both raw data and cluster summaries. Because the algorithms that employ this method do not process all of the data, they scale according to the size of the sampling rather than the entire dataset.

DISCUSSION & RESULTS

We evaluated the CBKM algorithm's performance against that of other algorithms in order to determine its performance (e.g., standard K-means and KM-HMR). A file is divided into several components equal to the block size specified for the cluster for a specific dataset (which is 64 MB by default). Numerous experiments on a variety of datasets were done to assess the quality and scalability of our suggested algorithms.

The datasets used in our investigation are listed in Table 1. Approximately 50,000 free ebooks have been downloaded through Project Gutenberg (PG). PG has over 3000 English publications published by 142 writers and is dedicated to creating and disseminating ebooks of largely public domain documents. Unstructured data is information that lacks a predetermined data model and is not structured in a predefined way. We chose this dataset because it lacks well defined observations and variables (rows and columns). Manually analysing the document is impossible. Clustering facilitates the categorization of texts based on their similarity. We implemented the approach on a subset of PG documents. To achieve successful clustering results, all of the proposed algorithms were run on document sets of varying sizes from the aforementioned dataset. Audio, music, entertainment, children, education, comics, crafts, finance, health, and

***e*ISSN** 1303-5150

markets are all areas of emphasis. This dataset was used to analyse and explore documents in order to do a more efficient cluster analysis of unstructured data.

Table 1 Datasets description

Dataset	Size (GB)	Description
DS_A: million song dataset	300	Collection of 53 audio features and metadata
DS_B: US climate reference network (USCRN)	200	Collected from 143 stations to maintain high quality climate observations
DS_C: Project Gutenberg	110	Includes over 50,000 free ebooks

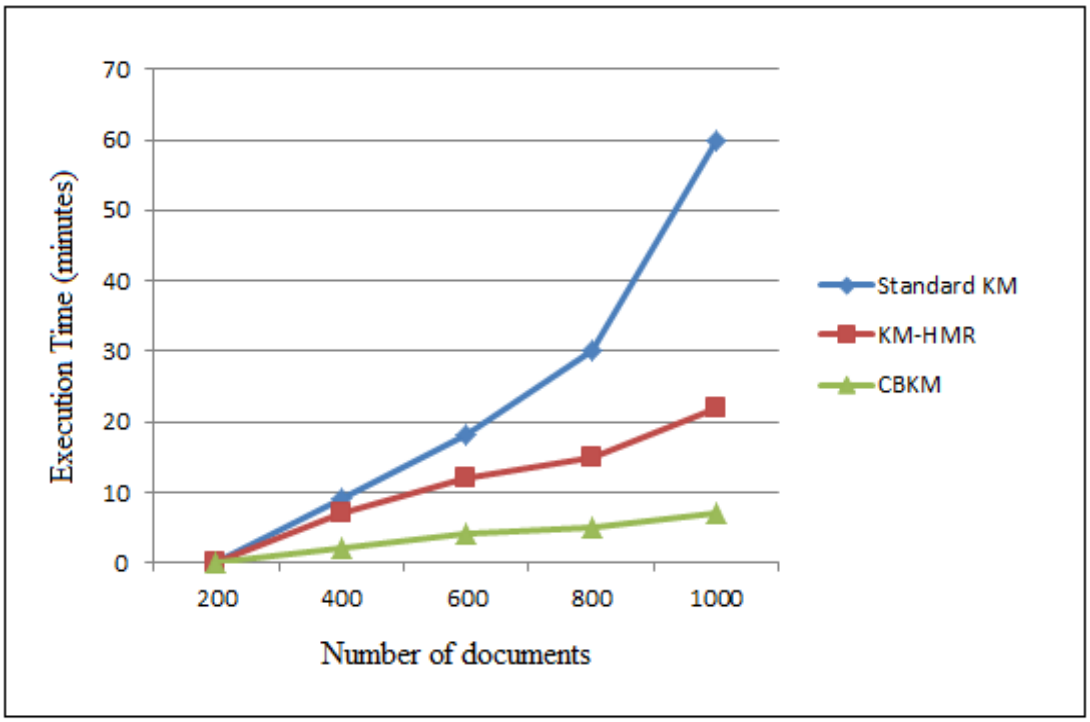

Figure 3 Comparison of execution times with number of documents

The execution timings of the suggested algorithms for document sets from the PG dataset are shown in Figure 3. With an increase in the number of documents to cluster, the time required for

standard K-means approaches the time required for the suggested solution. Parallel processing is critical when dealing with big amounts of data. We investigated only a part of the PG dataset in this experiment.

Table 2 Clustering results

Broad categories	Subcategories formed/total		
	KM-I2C	KM-HMR	Standard K-means
Digital audio	21/22	19/22	15/22
Space	26/28	17/28	14/28
Education	61/64	51/64	42/64
Entertainment	50/52	40/52	30/52

The clustering results for Project Gutenberg are shown in Table 2. Due to the fact that the proposed CBKM algorithm processes huge datasets over multiple machines, it takes less time to implement than the typical K-means algorithm. While KM-HMR employs the Euclidean distance as a measure of similarity, our proposed CBKM employs inter- and intra-clustering measures to produce efficient and high-quality clusters with high intra-cluster similarity and low inter-cluster similarity in order to extract valuable insights from large datasets.

CONCLUSION

Scalability of clustering algorithms is a relatively new topic that has developed as a result of the requirement for unsupervised learning in data mining applications. Due to their time or space complexity, the commonly used clustering methods cannot scale to larger datasets. This dilemma necessitates the development of novel ways for adapting frequently used clustering algorithms to current requirements. To address the drawbacks of standard sampling algorithms, a new collection algorithm is given to obtain the sample set by fully exploiting the rectangle domain, sampling domain, and sample point definitions. Future work will focus on automatically identifying parameters rather than having them supplied by users.

***e*ISSN** 1303-5150

REFRENCES :

1. B. Bozdemir, S. Canard, O. Ermis, H. Möllering, M. Önen, and T. Schneider, "Privacy-preserving density-based clustering," 2021.View at: Google Scholar

2. L. Wang, H. Wang, W. Zhou, and X. Han, "A novel adaptive density-based spatial clustering of application with noise based on bird swarm optimization algorithm," *Computer Communications*, vol. 6, 2021.View at: Google Scholar

3. Karwan Qader, Mo Adda and Mouhammd Al-kasassbeh (2017), "Comparative Analysis of Clustering Algorithms in Network Traffic Faults Classification", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 5, No.4, pp.6551-6563.

4. Yu Wang, Yang Xiang, Jun Zhang, Wanlei Zhou and BailinXie(2016), "Internet traffic clustering with side information", Journal of Computer and System Sciences, Elsevier, Vol. 80, No.5, pp.1021–1036.

5. M. J. Reddy and B. Kavitha, "Clustering the mixed numerical and categorical dataset using similarity weight and filter method," *International Journal of Database Theory and Application*, vol. 5, pp. 121–134, 2012.

6. Y. Wei, X. Zhang, Y. Shi et al., "A review of data-driven approaches for prediction and classification of building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 1027–1047, 2018.

7. M. Kumar, P. Chhabra, and N. K. Garg, "An efficient content based image retrieval system using BayesNet and *K*-NN," *Multimedia Tools and Applications*, vol. 77, no. 16, pp. 21557–21570, 2018.

8. A. Onan, S. Korukoğlu, and H. Bulut, "A hybrid ensemble pruning approach based on consensus clustering and multi-objective evolutionary algorithm for sentiment classification," *Information Processing & Management*, vol. 53, no. 4, pp. 814–833, 2017.

9. Luiz Fernando Carvalhoa, Sylvio Barbona, Leonardo de Souza Mendes and Mario Lemes Proença (2016), "Unsupervised learning clustering and self-organized agents applied to help network management", Expert Systems with Applications, Elsevier, Vol. 54, pp.29-47.

10. P. Dahiya and D. K. Srivastava, "A comparative evolution of unsupervised techniques for effective network intrusion detection in hadoop," in *Proceedings of the International Conference on Advances in Computing and Data Sciences*, pp. 279–287, Dehradun, India, April 2018.

11. Mohiuddin Ahmed and Abdun Naser Mahmood (2015), "Novel Approach for Network Traffic Pattern Analysis using Clusteringbased Collective Anomaly Detection", Annals of Data Science, Springer, Vol. 2, No.1, pp.111–130.

12. S. Mehrotra and S. Kohli, "Comparative analysis of *K*-means with other clustering algorithms to improve search result," in *Proceedings of the 2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, pp. 309–313, Delhi, India, October 2015.

13. Shital Salve and Sanchika Bajpai (2014), "Online stream mining approach for clustering network traffic", IJRET: International Journal of Research in Engineering and Technology, Vol. 3 No.2, pp.300- 304.

14. X. Zheng and N. Liu, "Color recognition of clothes based on *K*-means and mean shift," in *Proceedings of the Intelligent Control, Automatic Detection and High-End Equipment (ICADE)*, pp. 49–53, Beijing, China, July 2012.

15. Lin Guan-zhou, XIN Yang, NIU Xin-xin and JIANG Hui-bai (2010), "Network traffic classification based on semi-supervised clustering", The Journal of China Universities of Posts and Telecommunications, Elsevier, Vol. 17, Supplement 2, pp.84-88.

***e*ISSN** 1303-5150