IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 4 Issue 7, July 2017
ISSN (Online) 2348 – 7968 | Impact Factor (2016) – 5.264
www.ijiset.com

Performance Improvement in Clustering using Similarity Measurement method by categorizing the data items

Sunil Kumar Sharma[1], Neha Gour[2], Mukesh Kumar[3], Dr. Rajendra Gupta[4]
[1, 2] Research Scholar, AISECT University, Raisen
[3, 4] Assistant Professor, AISECT University, Raisen

Abstract

The cluster analysis make group of data objects based only on information found in the data which describes the objects and their relationships. The various evaluations on data clustering techniques reveal the need for effective methodology to capture the actual requirement of the user from while search the data over web browser. The goal of the study is to categorize the data of similar type in the form of cluster in effective way. That is possible by keeping the objects within a group that should be similar to one another and different from the objects in other groups. Better clusters are formed with greater the similarity (or homogeneity) within a group and the greater the difference between those groups. The paper is based on the clustering of data by effective method of similarity measurement. The obtained results show the better performance over the text data analysis.

Keywords : Clustering, k-means algorithm, modified k-means algorithm

1. Introduction

1.1 Overview of Clustering and its Algorithms

Clustering of data means that the task of dividing the data points or items into a number of groups such that the data points in the same groups should be more similar to other data points in the same group than those in other groups. In other words, the goal of clustering is to segregate groups with similar traits and assign them into different clusters.

Unlike classification, the person don't know what would be the clusters or by which attributes, the data would be clustered. As a result, someone who is knowledgeable in the business area must interpret the clusters. Often, it is required to modify the clustering by excluding variables that have been employed to group instances, because upon examination, the user identifies them as irrelevant or not meaningful. After finding the clusters that reasonably segment our database, the clusters may then be used to classify new data or data set. Some of the commonly known algorithms used to perform clustering of data include feature maps and K-means algorithm. Clustering with segmentation refers to the general problem of identifying groups that have common characteristics. The clustering is a way to segment data into groups that are not previously defined, whereas classification is a way to segment data by assigning it to groups that are already defined.

The clustering method or process which is used to make clusters for different data fields can be divided into two sub divisions :

- **Hard Clustering :** In this type of clustering, each data point may be either belongs to a cluster completely or not.
- **Soft Clustering :** In this clustering, in place of putting each data point into a separate cluster, a probability of that data point to be in those clusters is assigned.

The clustering algorithms can be divided into different groups based on its execution and their task performance. Every methodology follows different strategy and set of rules for defining the similarity among data items or data points. There are a number of clustering algorithms are in existence, but few of them are commonly used. The details of these clustering algorithms are as under:

- **Connectivity models or algorithm :** As its name, these algorithms are based on the notion that the data points closer in data space that exhibit more similarity to each other than the data items lying farther away. These algorithms can follow two approaches; In the first approach, they start with classifying all data points into separate clusters & then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of distance function is subjective. These models are very easy to interpret but lack scalability for handling big datasets. The examples of these models are hierarchical clustering algorithm and its variants.

- **Centroid methods :** These are iterative clustering algorithms in which the notion of similarity is derived by the closeness of a data point to the centroid of the clusters. K-Means clustering algorithm is a most frequently used algorithm that falls into this category. In these methods, the number of clusters are required at the end have to be mentioned beforehand, which makes it important to have prior knowledge of the dataset. These methods run iteratively to find the local optima.

- **Distribution models :** This type of clustering models are based on the notion of how probable is it that all data points in the cluster belong to the same distribution. These models often suffer from over fitting. A popular example of these models is Expectation-maximization algorithm which uses multivariate normal distributions.

- **Density Models:** These models search the data space for areas of varied density of data points in the data space. It isolates various different density regions and it assigns the data points within the defined regions in the same cluster.

1.2 Overview of k-means algorithm

K-means is the most widely used algorithm for clustering purposes and it is best-known algorithm for the partitioning-based clustering methods. Choosing the initial centroids randomly results in poor clustering. Instead of optimal clustering (global optimum), suboptimal clustering (local minimum) is obtained. One of the major problems with the K-means algorithm is that the empty clusters can be obtained if no points are allocated to a particular cluster during the assignment step. If this happens, then a strategy is needed to choose a replacement centroid.

The k-means algorithm works when the value of *k* is identified and the number of clusters we want to find. The basic purpose of using this algorithm is to find or pick *k* number of points as the "centroids" of the k clusters.

After that the following procedure is applied to mange centriod :

- For each point, input a point in the cluster to which the values is closest to it.
- Re arrange the cluster centroids
- Repeat the process until there is no changes in cluster between two successive iterations.
- To check the iterative improvement of the objective function, the sum of the squared distance from each point to the centroid of its cluster is calculated.

1.3 Analysis of k-means algorithm

The k-means algorithm is well known cluster analysis algorithm which can be applied on various data sets. But this algorithm which selects *k* items randomly from the population of data as initial centroids cannot always make proper or desired cluster. After modification and improvement in k-means algorithm can produce better results as compared to earlier.

During the analysis of k-means algorithm on the applied data items, following limitations has been found;

- It need to know *k* in advance always
- Can find out several *k*. It is observed that cluster tightness increases with increase in K. The best intra-cluster tightness occurs when k = n which mean that every point in its own cluster
- It tends to go to local minima that are sensitive to the starting centroids which try out multiple starting points
- The assumed clusters are spherical in vector space
- Sensitive to coordinate changes and weighting

Along with above finding, following are some suggestions for clustering of data;

- To make k-means algorithm more efficient, especially for dataset containing large number of clusters, it is needed to eliminate unnecessary distance of data points.
- In each iteration the k-means algorithm computes the distance between data items and all centers. It is computationally not feasible especially for massive datasets analysis.

2. Literature Review

A number of researchers have been done the study of k-means algorithm and proposed solutions for different clustering problems. Some of the similar study of data clustering is discussed herewith;

Framework to Extract Context Vectors from Unstructured Data using Big Data Analytics [1], this paper proposes a framework for computing context vectors of large dimensions over Big Data, trying to overcome the bottleneck of traditional systems. With an increase in data, the process of information extraction becomes difficult. To reduce the difficulty the author proposed a framework which is based on set of map and reducers, implemented on Apache Hadoop. The aim of this paper is to examine and propose a framework for computing context vectors of large dimensions over Big Data, trying to overcome the bottleneck of traditional systems.

Analysis of Hadoop Performance and Unstructured Data Using Zeppelin [2], in this paper to manage the huge data in terms of storage, sharing, exchange there are various process to be involved. The data from different sources is processed that should be error free and good quality. Maintaining of huge data and retrieving is difficult, based on cloud concept storing and retrieving of data from cloud is easy. This paper explores the Hadoop cluster on Amazon Elastic Cloud, perform the benchmark of data load time with traditional data processing application and Hadoop. Secondly they analyzed the unstructured data in Zeppelin with Spark.

EDSC: Efficient Document Subspace Clustering Technique for High-Dimensional Data[3], this paper presents an Efficient Document Subspace Clustering (EDSC) technique for high dimensional data that contributes to the existing system with respect to identification by eliminating the redundant data. The outcome of proposed system is able to perform cluster analysis of massive dataset in least duration of time. The proposed technique has been compared with existing system to find the effective document clustering process for high-dimensional data. The processing time of EDSC for subspace clustering is reduced by 50% as compared to the existing system.

An Overview on XML Semantic Disambiguation from Unstructured Text to Semi-Structured Data: Background, Applications, and Ongoing Challenges [4]. This is the survey paper which provides a concise and comprehensive review of the methods related to XML-based semi-structured semantic analysis and disambiguation. In first part the paper covers traditional word sense disambiguation methods for processing flat textual data. In Second section it describes and categorizes the disambiguation techniques developed and extended to handle semi-structured and XML data. In Third section paper describes current and potential application scenarios that can benefit from XML semantic analysis, including: data clustering and semantic-aware indexing, data integration and selective dissemination, semantic-aware and temporal querying, web and mobile services matching and composition, blog and social semantic network analysis, and ontology learning. The forth section discuss ongoing challenges and future directions, including: the quantification of semantic ambiguity, expanding XML disambiguation context, combining structure and content, using collaborative/social information sources, integrating explicit and implicit semantic analysis, emphasizing user involvement, and reducing computational complexity.

Extensible Query Framework for Unstructured Medical Data – A Big Data Approach[5], the paper describes extensible query based framework contains built-in modules but is flexible in allowing the user to import their own, making it extensible. The framework runs the modules in a Hadoop cluster making it efficient by utilizing the distributed computing capability of big data approach. The framework is tested through simulation. The framework allowed the user to run a different module using the previous output to further analyze the unstructured data it

also enabled the user to import a new module. The framework is feasible to handle unstructured medical data in an accurate, efficient and extensible manner.

Agglomerative Algorithm to Discover Semantics From Unstructured Big Data[6], this paper presents a graph model and an agglomerative algorithm for text document clustering. Author had tested the algorithm in three different data sets and presented working scenario and also compared with traditional clustering algorithms, such as k-means, principal direction division partitioning, Auto Class and hierarchical clustering.

The author write on the Review of Unstructured Architecture for Voice and Data Services in Mobile Communication [7]. This paper gives a detail review of hybrid routing protocols for mobile communication. By considering Proactive routing protocol and Reactive routing protocol discoveries the routing tables only for the destination that has traffic going through. The common problem associated with network is mobility management. To overcome this problem author proposed a model for higher degree of coverage with less traffic. Future work of this paper will focus on the reduced delay with increased packet delivery ratio and better control over paths.

The paper based on Big Data clustering validity[8] describes a new fuzzy validity index able to interpret the best partition of Big Data clustering. Called Fuzzy Validity Index with Noise-Overlap Separation (FVINOS), this new technique provides sufficient interpretation of the properties of the Big Data by detecting the overall geometric structure within and between clusters. The main contribution of FVINOS is to define a crisp and fuzzy clustering validation taking in account the structure of Big Data sets.

Feedback Analysis of unstructured data from Collaborative Networking a Big Data Analytics Approach[9], this paper discuss about the framework that evaluates the extent of effectiveness of conventional data mining algorithms on Big Data captured from education data in multiple social networking sites. The framework is designed using java. The proposed system attempts to understand the possibility of performing knowledge and discovery the process from Big Data using conventional data mining algorithms considering massive number of online educational data from social networking sites, the proposed system evaluates the effectiveness of performing data clustering and data mining for Big data.

Query Revision During Cluster Based Search on Large Unstructured Corporate [10], the paper designed a framework for unstructured data. The framework limits the view for cluster based re-ranking systems in the setting of e-Discovery, and studied the problem of query performance prediction fo. Framework helps user to make the decision of “whether to revise”. The framework consists of two components. First, they introduce a “limited view” which is a summary of a long cluster-based re-ranked list. Second, construct query predictors for this limited view, and provide their prediction as a second input to the user. This prediction is used to corroborate the inspection of the summary limited view. The proposed combination of a limited view and query performance prediction can assist search staff in determining whether to pursue an expensive query revision ornot, as well as save precious time by precluding inspections of lists with very few relevant documents during the early stages of commercially important applications such as eDiscovery.

Signature based Malware Detection for Unstructured Data in Hadoop[11], the paper proposed a fast string search algorithm based on map-reduce approach. The paper make use calm AV’s updated free virus signature database signature database. of the algrothem is Implemented on

different pattern matching algorithms to search for signatures in the content of files.For better performance in terms of execution time by optimizing the map-reduce algorithm based on release of new versions of Hadoop.

The author write on Probabilistic Clustering and Classification for Textual Data: an Online and Incremental Approach[12]. This paper proposes an incremental, online and probabilistic clustering algorithm for textual data, based on a mixture of Multinomial distributions. The model is compared with two existing models Restricted Boltzmann Machines (RBM) and use binary input variables. The advantage of the model is that only a single step over the training data is necessary to learn from it. As more texts are processed, the model improves its structure to better represent the data stream.

One more paper is based on the Method of Cloudizing Storing Unstrctured LiDAR Point Cloud Data by Mongo DB[13]. The paper first designs the architecture of cloudizing storage server cluster for LiDAR point cloud data in the light of distributed storage framework of MongoDB. Secondly, it forwards an organization model of point cloud data FLE chunks, in accordance with MongoDB distributed FLE system GridFS, based on which the point cloud fle chunks' parallel sharding is achieved. At the end the prototype system is designed and accomplished for point cloud data cloudizing storage, which is used to carry out experiments on accessing the point cloud data in GridFS, and the results approve that the cloudizing storage proposed in this paper performs better than local file system in the accessing point cloud data.

The paper which is based on the Transfer Learning Approach for Learning of Unstructured Data from Structured Data in Medical Domain[14] utilizes a bisecting k-means algorithm for the purpose of disease prediction. author proposed a model for identifying more relevant disease using readings mentioned in patient's pathology lab test report. This model is influenced by clustering and unsupervised transfer learning. The paper also demonstrate the effectiveness of our model using patient pathology lab report dataset and dataset used for storing different test names hemoglobin, sugar, etc.) of four diseases (Diabetes, Lipid profile cholesterol, Liver profile and Kidney profile). The main aim is to improve performance of the system by transferring knowledge, learned in one or multiple source tasks and use the same to improve learning in a related target task.

GPU enhanced parallel computing for large scale data clustering[15] shows the clustering hundreds of documents on a workstation quickly is an unsolved research problem but the GPU may provide hope. Using the CUDA platform from NVIDIA, they developed a Multiple Species Data Flocking implementation to be run on the NVIDIA GPU. The paper basically focused on single GPU data clustering solution. In the further study, more applications will adopt the CPU+GPU computing model to reduce the system computing time.

An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques[16] is proposed by author. This paper combines IBM-LS-AUM techniques to mesh adaption. The idea is to conserve the simplicity of the embedded approaches for grid generation process and overcome the difficulty of wall treatments by using mesh adaptation. The paper proposed four test cases by demonstrating the ability of the proposed method to obtain an accurate solution along with an accurate wall treatment even when the initial mesh does not contain any point on the level-set 0. At the end of the paper the author explains the accuracy of boundary layer by this mesh adaptation

technique with level set, in future the author investigate IBM-LS-AUM methods for moving bodies.

In the paper 'Clustering nominal data using unsupervised binary decision trees: Comparisons with the state of the art methods[17]' the author proposed an extension of CUBT (clustering using unsupervised binary trees) to nominal data. They provided and justify some guidelines and heuristics to tune the parameters in CUBT. Extensive comparisons are done with other approaches using simulations, and two examples of real datasets applications are discussed.

A comparative study of dimensionality reduction techniques to enhance trace clustering performances[18], the paper discusses the effect of applying dimensionality reduction (preprocessing) techniques on the performance of trace clustering. The experimental study includes three popular feature transformation techniques; singular value decomposition (SVD), random projection (RP), and principal components analysis (PCA), and the state-of-the art trace clustering in process mining. Combination of K means clustering with SVD or random projection has the best fitness value for simple event logs, According to case study the combination of K means with PCA is more suitable for complex event logs.

Beyond the hype: Big data concepts, methods, and analytics [19], the paper gives a consolidated description of big data by integrating definitions from practitioners and academics. The paper is based on the analytic methods used for big data analysis. A particular distinguishing feature of this paper is its focus on analytics related to unstructured data, which constitute 95% of big data. The paper also highlights the need to develop appropriate and efficient analytical methods to leverage massive volumes of heterogeneous data in unstructured text, audio, and video formats. It reinforces the need to devise new tools for predictive analytics for structured big data.

EPLS: A novel feature extraction method for migration data clustering[20], in this paper a novel numerical feature extraction method EPLS is proposed. The EPLS model includes (1) Mode Decomposition in which EEMD algorithm is applied to the aggregation dataset; (2) Dimension Reduction is carried out for a more significant set of vectors; (3) Least Squares Projection in which all testing data are projected to the obtained vectors. The EPLS feature extraction method can achieve high performance compared with several different clustering methods and distance measures

Clustering time-stamped data using multiple nonnegative matrices factorization[21]. In this paper, an approach for clustering time-stamped data and discovering the evolutionary trends of the clusters by using Multiple Nonnegative Matrices Factorization (MNMF) with smooth constraint over time. To utilize time-stamped information in the clustering process, an extra object-time matrix is constructed in our proposed method. Experimental results on real data sets demonstrate that the proposed approach outperforms the comparative algorithms with respect to Fscore, NMI or Entropy.

Improving data partition schemes in Smart Grids via clustering data streams [22]. The purpose of this paper is to apply unsupervised learning techniques to enhance the performance of data storage in Smart Grids. The improved eXtendedClassifier System for Clustering (XCSc) algorithm to present a hybrid system that mixes data replication and partitioning policies by means of an online clustering approach. Conducted experiments show that the proposed system outperforms previous proposals and truly fits with the Smart Grid premises.

A Survey On Clustering Techniques For Mining Big Data[23] presents a theoretical overview of current clustering techniques used for analyzing big data. Clustering process includes different patterns like classification, grouping, exploratory pattern analysis, machine learning, document retrieval, image segmentation and decision making. The key objective of this paper is to give general over view of general data clustering categorizations for big data.

A comparative study of efficient initialization methods for the k-means clustering algorithm[24], the paper gives an overview of methods with an emphasis on their computational efficiency. Eight commonly used linear time complexity initialization methods are compared on a large and diverse collection of data sets. Experimental results are analyzed using nonparametric statistical tests and provide recommendations for practitioners. The author compared eight commonly used linear time initialization methods on a large and diverse collection of real and synthetic data sets using various performance criteria are discussed.

Further clustering Techniques have been discussed with brief survey of different clustering algorithms. In this survey, review of different clustering techniques of data mining is discussed and focuses on the clustering basics, requirement, classification, problem and application area of the clustering algorithms. In this paper different clustering technique are compared and summarized. It also includes the classification of clustering techniques and their respective algorithms with the advantages and disadvantages.

3. Methodology

There are a number of clustering methods which can be classified into different categories, such as, Partitioning methods, Hierarchical methods, Density-based methods, Grid-based methods and model based methods. Each of these methods handles some of the issues related to clustering. However, in spite of the good properties of these methods, it is not favorable for very large-scale data sets because the complexity of these methods is highly dependent on the size of a dataset.

K-means and K-medoids are widely used simplest partition based unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters).

In this study, we have tried to put our strategy in enhancing the partition base clustering algorithm to improve accuracy and generate better and stable cluster and also tried to reduce its execution time and efficiently scaled a large data set. For this, we have used the concept of “initial centroids” which has proved to be a batter option for clustering. This new approach is analytically compared with the existing partition methods.

The procedure of the implementation in k-means algorithm represents overall clustering formation. The processing is depicting in Figure 1. The first step define the number of clusters taken in the study. Then the random number of objects a selected for the clustering. After selection of cluster centroids are calculated (i.e. median) and calculate the distance for each object with Centroid (Grouping based on minimum Distance). If there is no object move from one group into any other group, then the process is terminated otherwise the process is repeated until the final cluster is created.

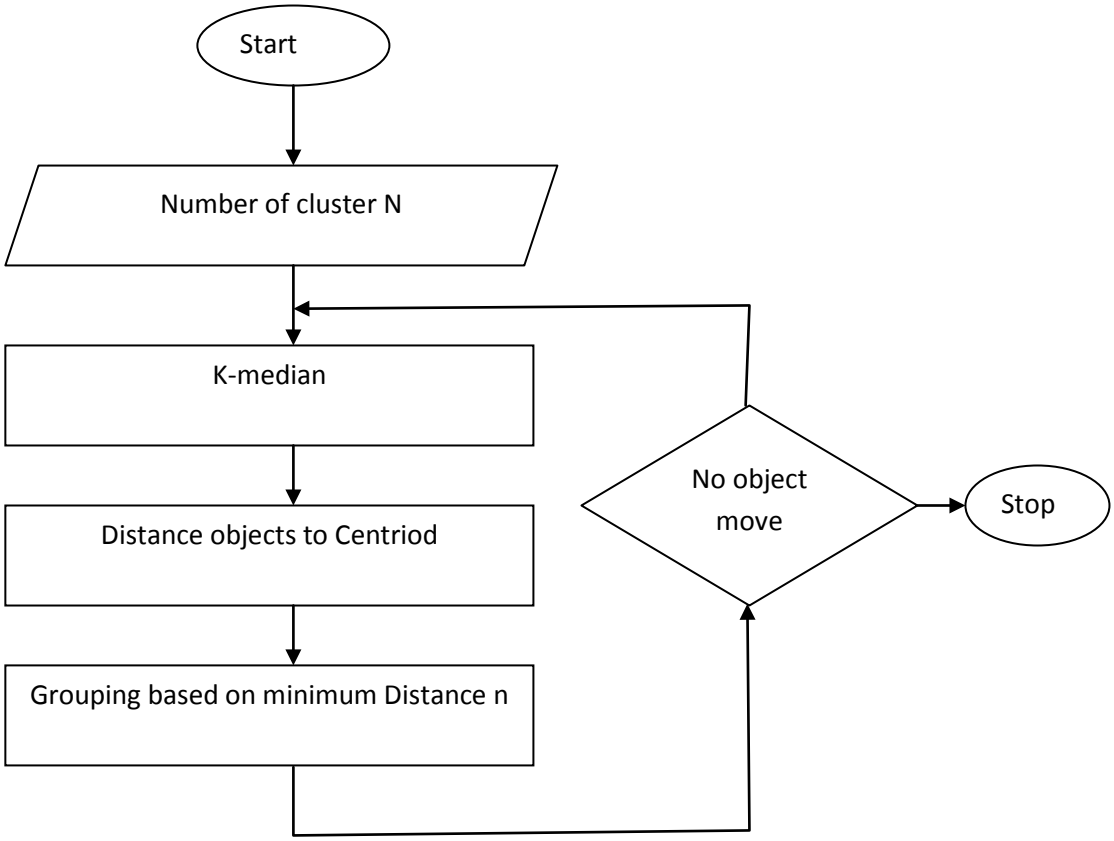

Figure 1 : Flowchart for Efficient k-mean Algorithm

3.1 Clustering Algorithm design

- Instead of selecting initial centroids randomly for the stable cluster, the initial centroids are determined systematically.
- Euclidean distance is calculated between each data point and selects two data-points between which the distance is the shortest and form a data-point set which contains these two data-points, then we delete them from the population.
- Find out nearest data point of this set and put it into new set.
- Calculate the mean value of each set that become initial centroid

3.2 Cluster Validation

It is difficult to identify that whether the clusters generated are of meaningful or just an artifact of an algorithm. Each clustering algorithm divides the given dataset into number of partitions without worrying about whether there exists any structure or not. Moreover, different clustering algorithm generates different result for the same dataset, and even some algorithm generates different result for different result for different set of parameters or different order of input data. Therefore there must be some evaluation standards and criteria to provide the user with the degree of confidence for the clustering results derived from the used algorithm.

Generally, there are three methods of validating criteria are used :

External indices: which is based on prior knowledge and used as a standard to validate clustering solutions.

Internal indices: which are independent of prior knowledge. They examine the clustering structure directly from the original data.

Relative criteria: compares different clustering structure to decide which one may best reveal the characteristics of the objects.

Instead of initial centroids that are selected randomly for the stable cluster, the initial centroids are determined systematically. It calculates the Euclidean distance between each data point and selects two data-points between which the distance is the shortest and form a data-point set which contains these two data-points, then it is deleted them from the population. Now find out nearest data point of this set and put it into new set. The numbers of elements in the set are decided by initial population and number of clusters systematically. These ways find the different sets of data points. Numbers of sets are depending on the value of k. Now calculate the mean value of each set that become initial centroid of the proposed k mean algorithm.

3.3 Algorithm for implementation in k-means algorithm

Input:

$D = \{d_1, d_2, ..., d_n\}$ // set of n data items

K // Number of desired clusters

Output:

A set of k clusters

Steps:

1. Set the variable p = 1
2. Compute the distance between each data point and all other data-points in the data set D
3. Find the closest pair of data points from the set D and form a data-point set Am (1<= p <= k) which contains these two data- points, Delete these two data points from the set D
4. Find the data point in D that is closest to the data point set Ap, Add it to Ap and delete it from D
5. Repeat step 4 until the number of data points in Am reaches 0.75*(n/k)
6. If p < k+1, then p = p+1, find another pair of data points from D between which the distance is the shortest, form another data-point set Ap and delete them from D, Go to step 4
7. For each data-point set Am (1 <= p <= k) find the arithmetic mean of the vectors of data points Cp(1 <= p <= k) in Ap, these means will be the initial centroids
8. Compute the distance of each data-point di (1<=i<=n) to all the centroids cj (1 <= j <= k) as d (di, cj)
9. For each data-point di, find the closest centroid cj and assign di to cluster j
10. Set Clustered [i++] = j; // j:Id of the closest cluster
11. Set Nearest_Dist [i++] = d(di, cj)
12. For each cluster j (1 <= j <= k+1), recalculate the centroids
13. Repeat

14. For each data-point di
    14.1 Compute its distance from the centroid of the present nearest cluster
    14.2 If this distance is less than or equal to the present nearest distance,
    Else
        14.2.1 For every centroid Cj (1<=j<=k) Compute the distance (di, cj);
        End for
        14.2.2 Assign the data-point di to the cluster with the nearest centroid Cj
        14.2.3 Set ClusterId [i] = j
        14.2.4 Set Nearest_Dist [i] = d (di, cj);
        End for
15 For each cluster j (1 <= j<= k+1), recalculate the centroids**;** until the convergence Criteria is met.

4. Result and Discussion

The following graph shows the comparison between K-mean and efficient implementation in K-mean with Number of Cluster and Execution Time

Figure 2 : Comparison between K-mean and efficient approach K-mean with Number of Clusters and Execution Time (in milliseconds)

Number of Records	Time taken to execute (In millisecond) K-Mean Algorithms	Time taken to execute (In millisecond) Efficient K-Mean Algorithm
300	96341	62634
400	117372	85322
500	136726	104241
600	167202	123532

Table 1 : Comparison between K-Mean and Efficient approach algorithm with Number of Records and Execution Time

The following graph shows the comparison between K-mean and efficient approach K-mean with Number of Record and Execution Time

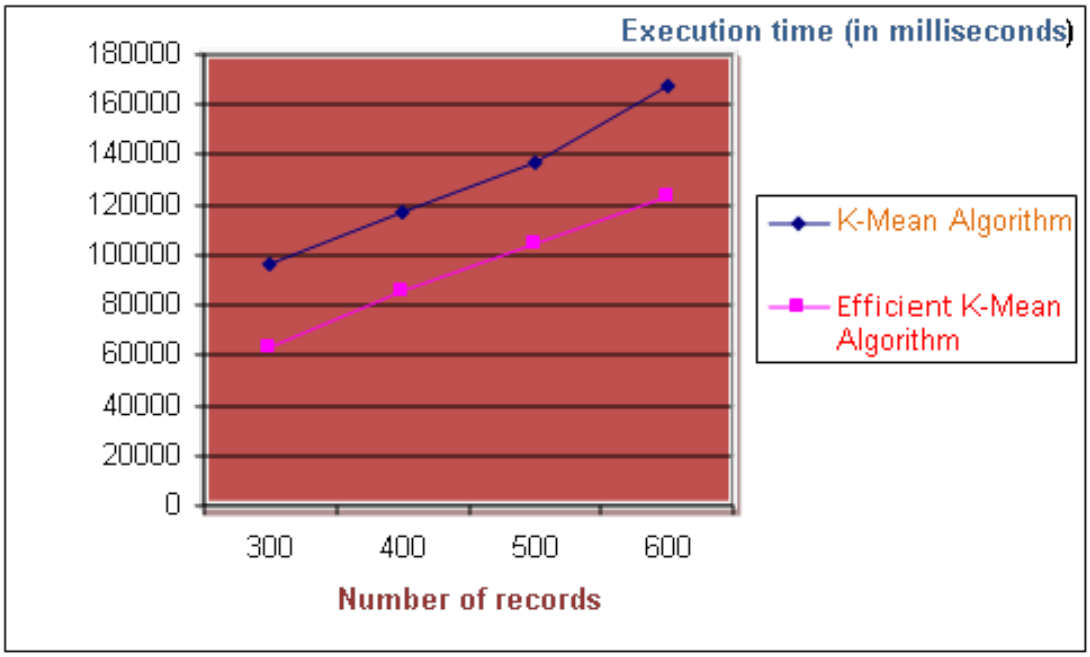

Figure 3 : comparison between K-mean and efficient approach K-mean with Number of Record and Execution Time (in milliseconds)

From the experimental results, following points has been observed;

- When the number of records is less, efficient approach of K-mean algorithm takes less execution time of computation then K-Mean Algorithm.
- When the number of clusters is more, then efficient approach of K-mean algorithm takes minimum time to execute than the K-mean.

5. Conclusion

The *k*-means algorithm is very popular algorithm for doing clustering and its analysis which is widely used in many applications but the standard algorithm which selects *k* objects randomly from population as initial centroids cannot always give a satisfactorily results. Selecting centroids from the training dataset by proposed algorithm can lead to a better clustering. The proposed algorithm is based on initial centroid which is selected systematically. It is a stable algorithm and generates accurate clusters. It takes less execution time for cluster generation as compared to existing *k*-mean algorithm. The implemented algorithm does not generate empty clusters so the working procedure and its execution become fast. Moreover it is more effective on noisy data.

References

[1] Tanvir Ahmad, Rafeeq Ahmad, Sarah Masud, Farheen Nilofer, "Framework to Extract Context Vectors from Unstructured Data using Big Data Analytics", Pages: 1 -6, DOI:10.1109/IC3.2016 IEEE 9th International Conference on Contemporary Computing (IC3), 2016

[2] Harleen, Naveen Garg, "Analysis of Hadoop Performance And Unstructured Data Using Zeppelin", Year: 2016, Pages: 1 -6, DOI : 10.1109/RAINS.2016.7764382, International Conference on Research Advances in Integrated Navigation Systems, April 06-07, 2016 IEEE

[3] Radhika K R, Pushpa C N, Thriveni J, Venugopal K R, "EDSC: Efficient Document Subspace Clustering Technique for High-Dimensional Data", , Year: 2016, Pages: 222 -226, DOI:10.1109/ICCTICT.2016.7514582, IEEE International Conference on Computational Techniques in Information and Communication Technologies (ICCTICT), 2016.

[4] Joe Tekli, "An Overview on XML Semantic Disambiguation from Unstructured Text to Semi-Structured Data: Background, Applications, and Ongoing Challenges", Year: 2016, Volume: 28, Issue: 6 Pages: 1383 - 1407, DOI: 10.1109/TKDE.2016.2525768, IEEE Transactions on Knowledge and Data Engineering, 2016

[5] SarmadIstephan, Mohammad-Reza Siadat, "Extensible Query Framework for Unstructured Medical Data – A Big Data Approach", Year: 2015, Pages: 455 - 462, DOI: 10.1109/ ICDMW.2015.67, 2015 IEEE International Conference on Data Mining Workshop (ICDMW)

[6] I-Jen Chiang, "Agglomerative Algorithm to Discover Semantics From Unstructured Big Data", Year: 2015 Pages: 1556 - 1563, DOI: 10.1109/BigData.2015.7363920,2015 IEEE International Conference on Big Data (Big Data)

[7] Aarti Rahul Salunke, Arun Natha Gaikwad, "Review of Unstructured Architecture for Voice and Data Services in Mobile Communication", Year: 2015,Pages 1-7, DOI: 10.1109/ICECCT.2015.7226190,2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)

[8] Mania Tlili, Tarek M. Hamdani, "Big data clustering validity", Year: 2014,Pages:348-352,DOI: 10.1109/SOCPAR.2014.7008031, 2014 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR)

[9] Ganeshayya Shidaganti, S. Prakash, "Feedback analysis of unstructured data from collaborative networking a Big Data analytics approach" year:2014, Pages:343-347, DOI:10.1109/CIMCA IEEE,International conference on circuits, communication, control and computing

[10] Vinay Deolalikar, "Cluster based search on large unstructured corpora", Year: 2014 Pages: 845 - 853, DOI : 10.1109/ BigData.2014.7004314, 2014 IEEE International Conference on Big Data (Big Data)

[11] Abhaya Kumar Sahoo, Kshira Sagar Sahoo, Mayank Tiwary, "Signature based malware detection for unstructured data in Hadloop", Year: 2014, Pages: 1-6, DOI:10.1109/ICAECC 2014, International Conference on Advances in Electronics Computers and Communications

[12] Thiago Fredes Rodrigues, Paulo Martins Engel, "Probabilistic Clustering and Classification for Textual Data: An Online and Incremental Approch",Year:2014, Pages:288-293,DOI:10.1109 /BRACIS.2014.59, 2014 Brazilian Conference on Intelligent Systems

[13] Wende Wang; Qingwu Hu, " The Method Cloudizing Storing Unstructured LiDAR Point Cloud Data by Mongo", Year: 2014, Pages: 1-5, DOI: 10.1109/ GEOINFORMATICS.2014.6950820, 2014 22nd International Conference on Geoinformatics.

[14] Nishigandha V. Wankhade, Madhuri A. Patey, "Transfer learning approach for learning of unstructured data in medical domain", Year:2013,Pages: 86 – 91, 2013 2nd International Conference on Information Management in the Knowledge Economy.

[15] Xiaohui Cui, Jesse St. Charles, Thomas Potok, "GPU enhanced parallel computing for large scale data clustering", Future Generation Computer Systems 29 (2013) 1736–1741, © 2012 Elsevier

[16] R. Abgrallc, H. Beaugendrea, C. Dobrzynski, "An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques", Journal of Computational Physics, Volume 257, Part A, 15 January 2014, Pages 83-101.

[17] Badih Ghattas, Pierre Michel, Laurent Boyer, "Clustering nominal data using unsupervised binary decision trees: Comparisons with the state of the art methods", Pattern Recognition, Volume 67, July 2017, Pages 177-185

[18] M. Song, H. Yang, S.H. Siadat, M. Pechenizkiy, "A comparative study of dimensionality reduction techniques to enhance trace clustering performances", Expert Systems with Applications, Volume 40, Issue 9, July 2013, Pages 3722-3737.

[19] Yunliang Chen, Fangyuan Li, Jia Chen, Bo Du, Kim-Kwang Raymond Choo, Houcine Hassan, "EPLS: A novel feature extraction method for migration data clustering", Journal of Parallel and Distributed Computing, Volume 103, May 2017, Pages 96-103

[20] Xiaohui Huang, Yunming Ye, LiyanXiong, Shaokai Wang, Xiaofei Yang, "Clustering time-stamped data using multiple nonnegative matrices factorization" Knowledge-Based Systems, Volume 114, 15 December 2016, Pages 88-98

[21] Andreu Sancho-Asensio, Joan Navarro, ItziarArrieta-Salinas, José Enrique Armendáriz-Íñigo, Virginia Jiménez-Ruano, AgustínZaballos, Elisabet Golobardes "Improving data partition schemes in Smart Grids via clustering data streams", Expert Systems with Applications, Volume 41, Issue 13, 1 October 2014, Pages 5832-5842

[22] Prachi Surwade1, Prof. Satish S. Banait 2 ," A Survey On Clustering Techniques For Mining Big Data", International Journal of Advanced Research in Science Management and Technology, Volume 2, Issue 2, February 2016,

[23] M. Emre Celebi, Hassan A. Kingravi, Patricio A. Vela "A comparative study of efficient initialization methods for the k-means clustering algorithm" Expert Systems with Applications, Volume 40, Issue 1, January 2013, Pages 200-210

[24] Kehar Singh, Dimple Malik and Naveen Sharma, "Evolving limitations in K-means algorithm in data mining and their removal", International Journal of Computational Engineering & Management, Vol. 12, April 2011, pp. 105-110