JCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-6, Issue-12, Dec 2018 E-ISSN: 2347-2693

A Revised and efficient K-means Clustering Algorithm

P. Jat[1*], K. Jain[2]

[1, 2]Computer Science and engineering, College Of Technology and Engineering, Maharana Pratap University of Agriculuture and Technology, Udaipur, India

[*]*Corresponding Author: Poojajat1823@gmail.com*

Available online at: www.ijcseonline.org

Accepted: 05/Dec/2018, Published: 31/Dec/2018

***Abstract*—**In digital era large volumes of data are generated by enterprises. Mining on this large volume of data provides valuable insights into user behaviors and helps to improve the business. Various Machine learning algorithms are proposed for data mining. Clustering is an important data mining algorithm for grouping the records and analyzing the data. K-means is a most used Clustering algorithm, but the time taken to cluster large volume of records is high. To reduce the clustering time many approaches are proposed in literature. In this work an improved K-means clustering is proposed which is able to reduce the clustering time.

***Keywords*—** K-means, Clustering, Centroids

I. INTRODUCTION

Data mining is emerging as a new fundamental research area with applications in various domains of engineering, science, medicine, business and education. Extraction of meaningful information and knowledge from unstructured data is facilitated with use of data mining.

Clustering is unsupervised learning technique with the aim of grouping set of objects into subsets or clusters. Clusters created are coherent internally but different from other clusters. Existing clustering algorithms are categorized to following types

1. Partitioning clustering
2. Hierarchical clustering
3. Density based algorithms
4. Grid based methods
5. Model based method

Partitioning approach split the dataset to flat K partition. Hierarchical Clustering creates hierarchical partitions with each cluster further split to sub clusters. Density based clustering creates group based on spatial distribution of the data. Grid based clustering splits objects to finite grids. Model based methods create a model for each cluster and fits data to any of the models.

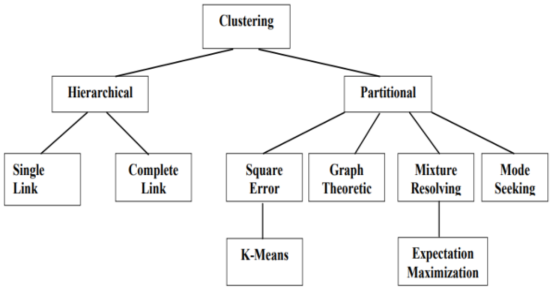

Figure 1:Classification Of Clustering

K-means is a flat clustering algorithm coming under the category of Partitioning approach. The objective of K-means is to minimize the average squared distance of objects from their cluster centers where cluster center is defined as the mean or centroid of the objects in a cluster: The squared distance of each data point from its centroid is called as Residual sum of Squares (RSS). K-means starts with selecting K random objects as cluster centers and move the centers around the space to minimize the RSS.

K-Means clustering intends to partition number of objects into k clusters in which each object belongs to the cluster with the nearest mean. This method produces exactly k different clusters of greatest possible distinction. The best number of clusters k leading to the greatest separation (distance) is not known as a priori and must be computed from the data. The objective of K-Means clustering is to minimize total intra-cluster variance, or, the squared error function:

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - C_j \right\|^2$$

Where $\left\| x_i^{(j)} - C_j \right\|^2$ is the distance function J is the objective function. K is the number of cluster. n is the number of cases . x_i is the case i and C_j is the centroid for cluster j.

The pseudo code of K-means algorithm is

```
Input:  K (the number of clusters),
        D (a set of lift ratio)
Output: a set of k clusters
Method:
Arbitrarily choose k objects from D as the initial cluster centres;
Repeat:
    1. (re)assign each object to the cluster to which the object is
       the most similar, based on the mean value of the objects in
       the cluster;
    2. Update the cluster means, i.e., calculate the mean value of the
       objects for each cluster.
Until no change;
```

K-means clustering algorithm has some problems

There is no guideline on choosing efficient number of clusters to be formed for a dataset. K-means selects initial centroids randomly. Due to improper selection of centroids, the number of iteration for completion of clustering increases or within configured iteration K-means does not converge. Also due to random cluster centroids, the results are not consistent.

In this paper, an improved K-means clustering algorithm is proposed. The optimal number of clusters is found using calinski-harabasz index and clustering is done on a sorted dataset to minimize the number of movements and efficient selection of new cluster heads. With the reduction in number of movement and cluster head selection principle, the time taken for clustering is reduced and the clusters exhibit higher coherence within clusters. The proposed solution was tested against various dataset available from public repository (UCI machine learning repository) and its effectiveness in terms of time and coherence is compared with other K-means variants.

II. Related Work

The existing solutions for improvement in K-means clustering algorithm is reviewed in this section.

In [1] authors proposed a Huffman tree based solution to select the initial cluster centers and normal k means algorithm is executed with those cluster centers. When multiple attributes influence the dimension contribution rate in the data set the Huffman based cluster selection performance reduces.

In [2] ranking on a particular attribute and clustering on that attribute was proposed to reduce the clustering time. Even though this approach works for certain applications, the analysis on combined attributes fails and many real world applications are based on combined attribute based analysis.

In [3], authors proposed a heuristic method to find the initial centroids. By averaging the attribute of each data point in the dataset, initial centroids are generated. For multi-dimensional attributes weight factor is associated with each attribute based on degree of variance. The algorithm complexity increases exponentially with the increase in the number of attributes.

In [4], author improved the K-means applying noise data filtering. Before clustering, preprocessing is done on the data to remove noise data which affects the clustering efficiency. By this way the clustering quality was improved. The impact on efficiency is not so high in this approach.

In [5] author proposed a improvement method to remove noise and outlier which impacts K-means clustering effectiveness. Even though RSS of the clusters are reduced, the clustering time is high in this approach.

In [6], authors proposed a ElAgha initialization algorithm that generates initial clusters depending on the overall shape of the data. It finds the boundaries of data points and divides the area covered by points into two dimensional grid. The initial centroids selected allows K-means to converge to a local minimum. But the approach is data specific.

In [7], authors proposed a Kd tree based algorithm for initial centroids selection. It performs density estimation at various points to choose K-seeds.

In [8] author proposed a density estimation based centroids selection. In this method distances between data object are computed and the density parameter of every data sample is counted. Then, the data samples with the biggest density parameter are chosen as the initial clustering centers to find k initial clustering centers. The algorithm does not work well for spare datasets.

In [9] authors proposed a balanced K-means algorithm. The idea is to normalize all the feature values of dataset before clustering. All the feature values are projected into fix range so it can reduce the number of iterations in the standard K-means clustering. This approach reduces the clustering time at the cost of decreased coherence in clusters.

In [10] authors proposed a new Competitive K-means to reduce the inconsistent results in K-means++. Authors provided an efficient map reduce implementation to reduce the clustering time. The algorithm improves cluster analysis accuracy and decreases variance.

There are some limitations in the existing work. In existing algorithms, Data sets are specific. The time complexity is high. The accuracy and efficiency of existing work is not so high. The algorithms are not efficient for spare data. There are also other boundaries of the subsisting work like multi attribute influence is not considered, decreased coherence and increases clustering time.

III. METHODOLOGY

The proposed work consist of three parts

1. Estimation of Efficient number of clusters
2. Selection of centroids
3. Clustering with initial K centroids

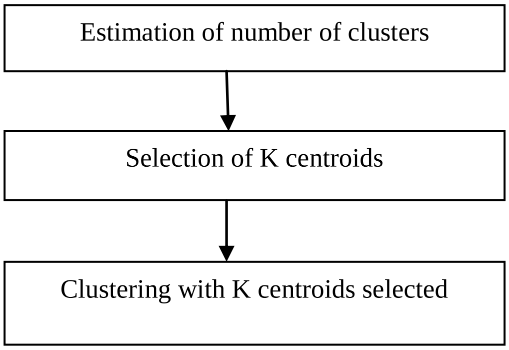
Estimation of number of clusters
Selection of K centroids
Clustering with K centroids selected

Figure.2: Flow of proposed work

The first step in the proposed work is to estimate the number of clusters. Once number of clusters is estimated, the centroids are selected from the dataset in a consistent way, so that for any data set irrespective on how many runs, the clustering result is consistent.

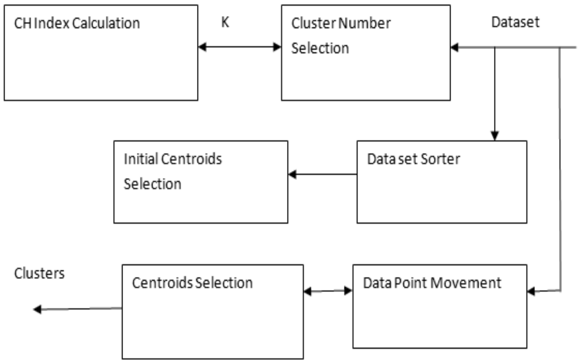

Figure.3: Block diagram proposed work

The modules in the system are
CH Index Calculation: This module calculates the Calinski-Harabasz index on the dataset.
Cluster Number Selection: This module invokes CH index calculation module for different values of cluster and gets the maximum value of cluster number which is efficient for the given dataset.

Data set Sorter: This module sorts the dataset with respect to a origin point using New modified sorting algorithm[10].
Initial Centroids Selection: This module selects the initial centroids by splitting the sorted data set to partitions and selecting the median from each partition as centroids.
Data Point Movement: This module moves the data points to cluster of closest centroids.

Centroids Selection: Once data point are moved, reselection of centroids is implemented in this module.

1. Cluster Estimation

The number of clusters is estimated using Calinski-Harabasz index.
Calinski-Harabasz index is based on the ratio between cluster scatter matrix (BCSM) and within cluster scatter matrix (WCSM).

$$CH_k = \frac{BCSM}{k-1} . \frac{n-k}{WCSM}$$

Where n is the total number of points and k is the number of clusters.

BCSM is calculated as

$$BCSM = \sum_{i=1}^{k} n_i \,.\, d(z_i . z_{tot})^2$$

Where z_i is the center of cluster c_i and n_i , the number of points in c_i .

WCSM is calculated as

$$WCSM = \sum_{i=1}^{k} \sum_{x \in c_i} d(x, z_i)^2$$

where x is a data point belonging to cluster c_i . For efficient clustering, BCSM must be maximized and WCSM must be minimized. The resulting CH value for this condition is the efficient number of clusters.

From the data set, BCSM and WCSM is evaluated for different partitions size and the partition size and from this the maximum value of CH is selected as the optimal number of cluster for that dataset.
The Calinski-Harabasz index is calculated for different cluster size values for IRIS dataset and plotted below

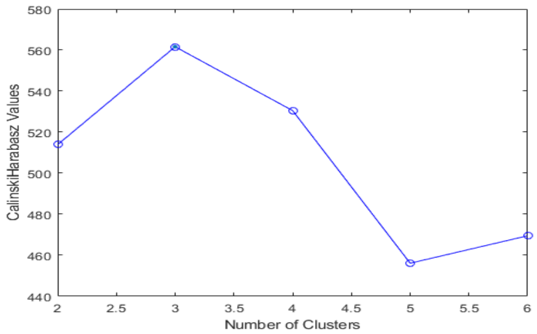

Figure.4: different cluster size values for IRIS dataset

The plot shows that the highest Calinski-Harabasz value occurs at three clusters, suggesting that the optimal number of clusters is three. To test if the returned result from the Calinski-Harabasz is optimal, clustering is done on the IRIS data set with the results from CH and the plotted below

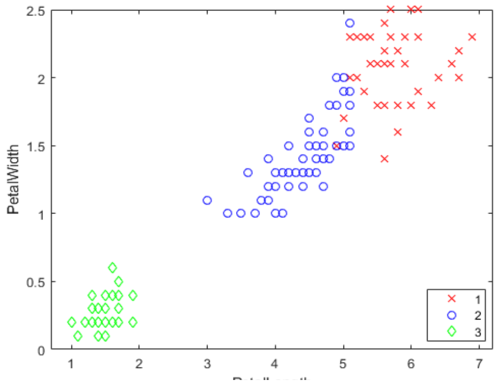

Figure.5: different cluster size values for IRIS dataset

From the results, it can seen the clusters have high coherence and thereby proves that the Calinski-Harabasz has returned the optimal value.

2. Selection of centroids

The distance of each data point in dataset to the origin point is calculated and based on this distance the data points in the dataset are sorted. Sorting is done using “A new modified sorting algorithm”. In first part values are divided into three parts positive, negative and zero. In second part of algorithm negative and positive number are sorted, but without repeat number [10].

After sorting, the sorted dataset is then split to CH number of partitions (CH is found using Cluster Estimation). From each partition, the middle data object is selected as centroid.

3. Clustering

The clustering process is as follows

1. Measure the Euclidean distance for each data point to the K centroids.
2. Assign the each data point to the cluster of closest centroid
3. Recalculate the centroids and repeat the procedure till the stopping criterion is met.

The stopping criterion is when there is no change in centroids.

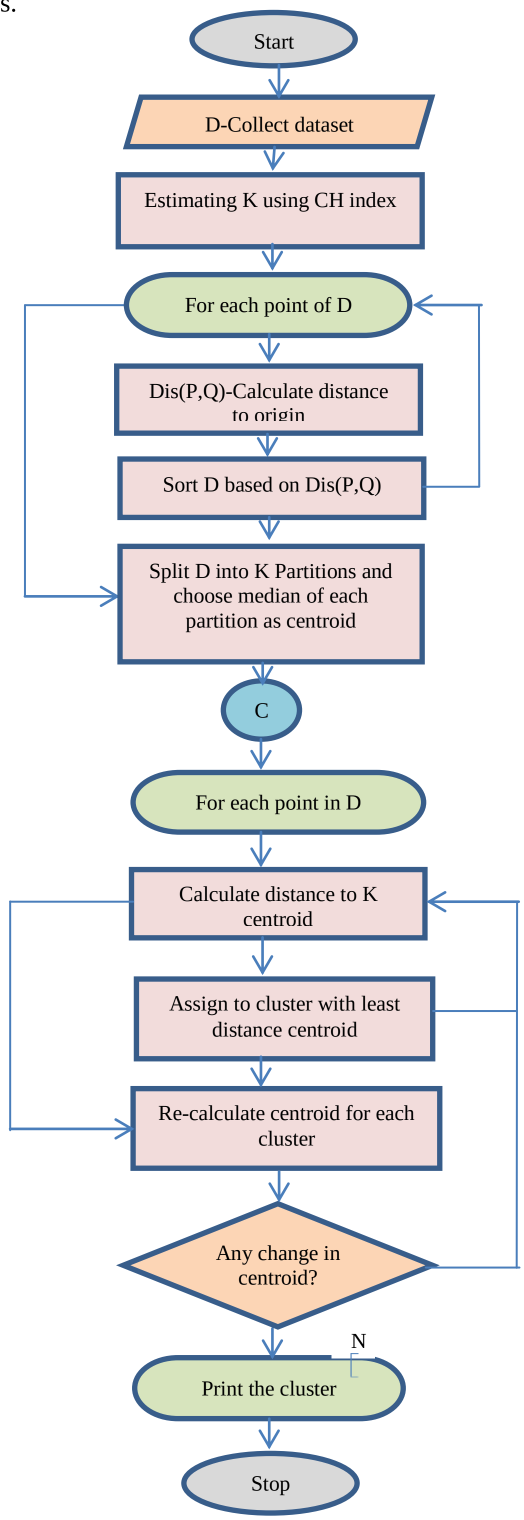
Start
D-Collect dataset
Estimating K using CH index
For each point of D
Dis(P,Q)-Calculate distance to origin
Sort D based on Dis(P,Q)
Split D into K Partitions and choose median of each partition as centroid
C
For each point in D
Calculate distance to K centroid
Assign to cluster with least distance centroid
Re-calculate centroid for each cluster
Any change in centroid?
N
Print the cluster
Stop

Figure.6: Flowchart of proposed work

Due to the proposed steps, the number of data movement is reduced. Due to movement reduction algorithm speeds up increases. Since there is no randomness in centroids selection, there is no inconsistency in the clustering results.
The clustering results achieved in IRIS dataset for the proposed is given below figure.7.

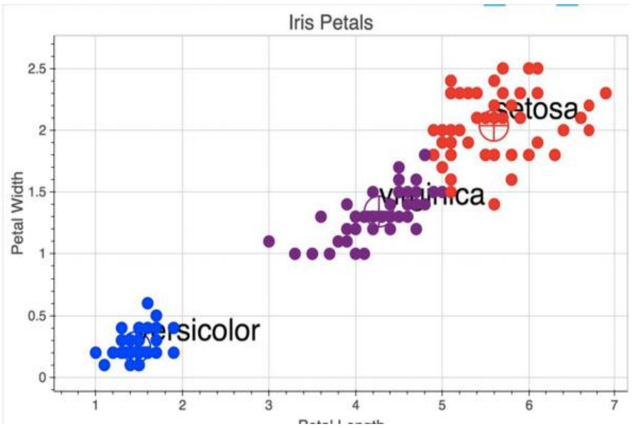

Figure.7: Clustering results for IRIS dataset

From the plot, it can seen that the clusters generated have high coherence compared to existing solution.

IV. Results and Discussion

The random data sets required to evaluate the proposed methodology was taken from UCI Machine learning repository[12]. For comparison Traditional K-means, Shina improved K-means, Modified K-means are used. Accuracy of clustering and time to clusters are the parameters evaluated for comparing the solutions.

IRIS Dataset

The data set contains 3 classes of 50 instances each, where each class refers to a type of iris plant.
The results and comparison of accuracy for the IRIS dataset across the solutions is given below figure.8.

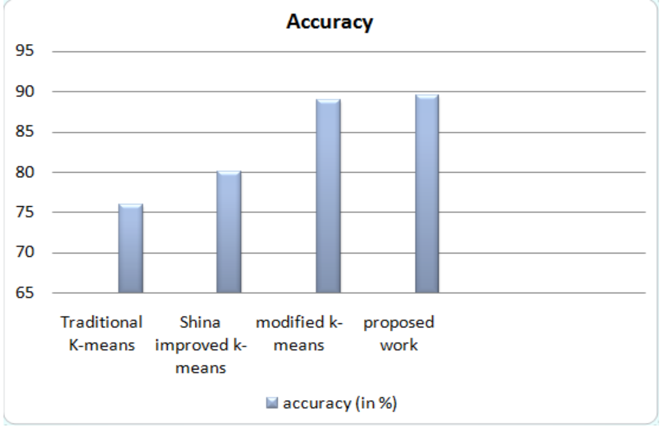

Figure.8: Accuracy comparison Graph for IRIS dataset

The results and comparison of time for the IRIS dataset across the solutions is below figure.9.

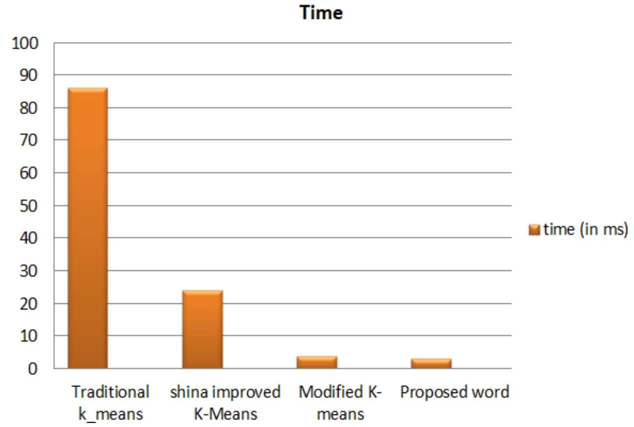

Figure.9: time comparison Graph for IRIS dataset

From the results, it can been that the proposed work has 13.5% more accuracy than traditional K-means and has able to reduce the clustering time from 86 milli seconds to a very low value of 3 milli seconds.
The comparison of results in terms of number of iterations for a cluster size of 3 is given below table.1.

Table 1: Iteration comparison on Iris dataset

Methods	Number of iterations
K-Means	51
Shina improved K-Means	20
Modified K-Means	7
Proposed Solution	4

WINE Dataset

The dataset has 13 attributes with 178 instances with attributes gathered from chemical analysis on wine.
The results and comparison of accuracy for the WINE dataset across the solutions is below figure.10.

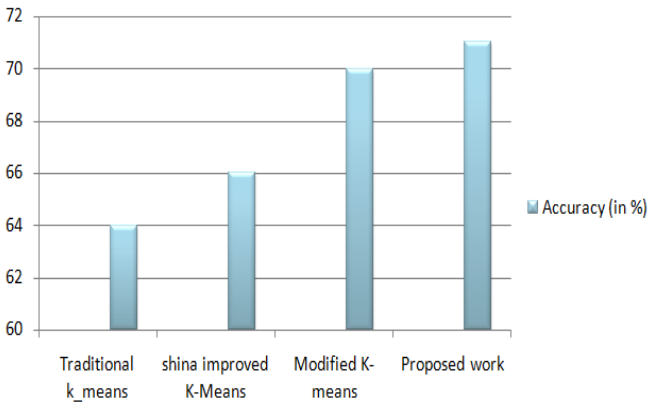

Figure.10: Accuracy comparison Graph for WINE dataset

The results and comparison of TIme for the WINE dataset across the solutions is below figure.11.

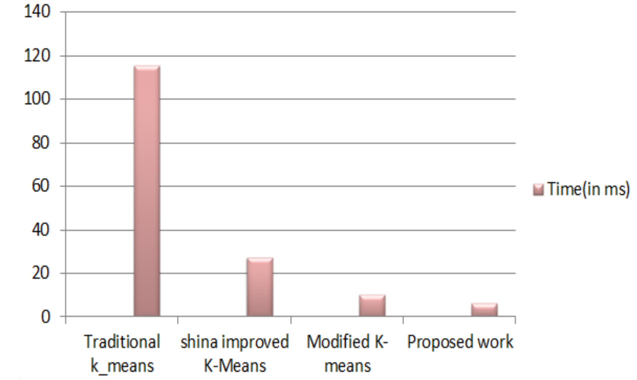

Figure.11: Time comparison Graph for WINE dataset

From the results, it can been that the proposed solution has 7% more accuracy than traditional K-means and has able to reduce the clustering time from 115 milli seconds to a very low value of 6 milli seconds.

The comparison of results in terms of number of iterations for a cluster size of 6 is given below table.2.

Table 2: Iteration comparison on wine dataset

Methods	Number of iterations
K-Means	57
Shina improved K-Means	33
Modified K-Means	20
Proposed Solution	11

Ecoli Dataset

The dataset has 8 attributes with 336 instances with attributes collected based on protein analysis on bacteria.

The result is given below figure.12 and from the result, it can be seen that the proposed algorithm takes lesser time than original K-means.

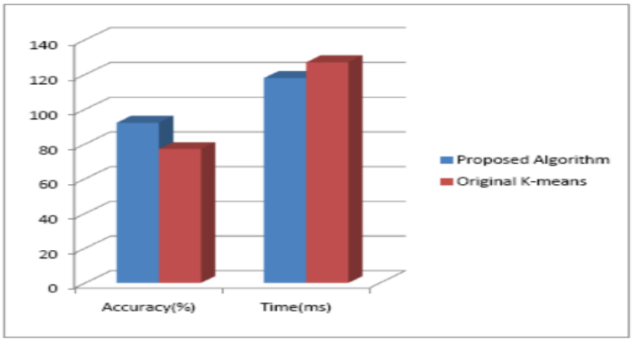

Figure.12: Accuracy comparison Graph for Ecoli Dataset

Brain Cancer Dataset

The dataset has 32 attributes with 569 instances. The data is collected from features extracted from the nuclei part of breast cancer images. The result is below figure.13.

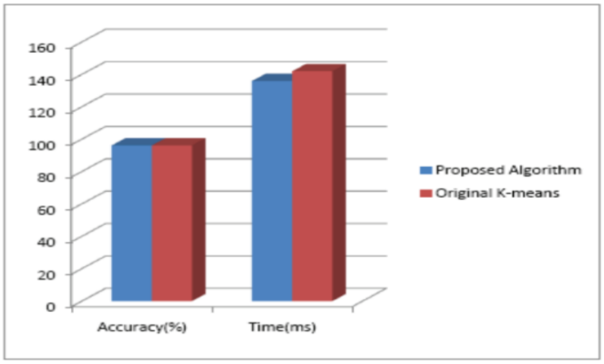

Figure.13: Accuracy comparison Graph for Brain cancer Dataset

V. Conclusion and Future Scope

In this work, improvement over K-means algorithm was proposed. Efficient number of clusters for a dataset is found using calinski harabasz index and then initial centroids were selected based on sorting the data points with respect to a origin point. The proposed solution was able to reduce the number of iterations and thus clustering time is reduced. The clusters created were highly cohesive within clusters. The proposed solution was tested against different datasets from machine learning repository and was able to get more than 7% accuracy and was able to reduce the execution time by 20 times when compared to traditional K-means. As a future work, outlier and noise removal on pre-processing stage must be evaluated to find out the gain in clustering accuracy and clustering time.

References

[1] Wang Shunye “An Improved K-means Clustering Algorithm Based on Dissimilarity” 2013 International Conference on Mechatronic Sciences, Electric Engineering and Computer (MEC)Dec 20-22, 2013, Shenyang, China IEEE.

[2] Navjot Kaur, Jaspreet Kaur Sahiwal, Navneet Kaur “EFFICIENT KMEANSCLUSTERING ALGORITHM USING RANKING METHOD IN DATA MINING” ISSN: 2278 – 1323 International Journal of Advanced Research in Computer Engineering & Technology Volume 1, Issue 3, May2012.

[3] Md. Sohrab Mahmud, Md. Mostafizer Rahman, and Md.Nasim Akhtar —”Improvement of K-means Clustering algorithm with better initial centroids based on weighted average” 2012 7th International Conference on Electrical and Computer Engineering 20-22 December, 2012, Dhaka, Bangladesh, 2012 IEEE.

[4] Juntao Wang & Xiaolong Su “ An improved K-Means clustering algorithm” 2011 IEEE.

[5] Mohamed Abubaker, Wesam Ashour, "Efficient Data Clustering Algorithms: Improvements over K-means", International Journal of Intelligent Systems and Applications, vol. 5, issue 3, pages 37-49, 2013.

© 2018, IJCSE All Rights Reserved

[6] Mohammed EI Agha, Wesam M. Ashour, " Efficient and Fast Initializtion Algorithm for K-means Clustering", LJ. Intelligent Systems and Applications, vol. 4, issue 1, pages 21-31, 2012

[7] Stephen J. Redmon, Conor Heneghan, " A method for initializing the K-means clustering algorithm using kd-trees", Journal Pattern Recognition Letters, vol. 28, issue 8, pages 965-973, 2007.

[8] Ling-bo Han, Qiang Wang, Zhengfeng Jiang etc..Improved k-means initial clustering center selection algorithm. Computer Engineering and Applications. 2010, 46(17):150–152.

[9] Wang, H., Qi, J., Zheng, W., & Wang, M. "Balance K-means algorithm. In Computational Intelligence and Software Engineering," Cise 2009 International Conference on, pp. 1-3, IEEE

[10] Idrizi F., Rustemi, A., & Dalipi F., (2017, June), Anew modified sorting algorithm: A comparison with state of the art. *In embedded computing (MECO) .20176th Mediterranean Conference on* (pp 1-6)IEEE.

[11] Esteves, R. M., Hacker, T., & Rong, C. "Competitive k-means, a new accurate and distributed k-means algorithm for large datasets" In Cloud Computing Technology and Science (cloudcom), 2013 IEEE 5th International Conference on ,Vol. 1, pp. 17-24.

[12]MerzCand Murphy P, UCI Repository of MachineLearningDatabases,Available:ftp://ftp.ics.uci.edu/pub/machine-learning-databases

© 2018, IJCSE All Rights Reserved