Inter Cluster Distance Management Model with Optimal Centroid Estimation for K-Means Clustering Algorithm

M.VIJAYAKUMAR[1], S.PRAKASH [2], R.M.S.PARVATHI [3]

[1]Department of Computer Science and Engineering, Sasurie College of Engineering
[2]Department of information Technology, Sasurie College of Engineering
[3]Department of Computer Science and Engineering, Sengunthar College of Engineering for Women
Tamilnadu, INDIA
Email : [1] tovijayakumar@gmail.com, [2] prakash_ant2002@yahoo.co.in

Abstract:- Clustering techniques are used to group up the transactions based on the relevancy. Cluster analysis is one of the primary data analysis method. The clustering process can be done in two ways such that Hierarchical clusters and partition clustering. Hierarchical clustering technique uses the structure and data values. The partition clustering technique uses the data similarity factors. Transactions are partitioned into small groups. K-means clustering algorithm is one of the widely used clustering algorithms. Local cluster accuracy is high in the K-means clustering algorithm. Inter cluster relationship is not concentrated in the K-means algorithm. K-means clustering algorithm requires the cluster count as the major input. The system chooses random transactions are initial centroid for each cluster. Cluster accuracy is associated with the initial centroid estimation process. The random transaction based centroid selection model may choose similar transactions. In this case the cluster accuracy is limited with respect to the distance between the centroid values. The proposed system is designed to improve the K-means clustering algorithm with efficient centroid estimation models. Three centroid estimation models are proposed system. They are random selection with distance management, mean distance model and inter cluster distance model. Cosine distance measure and Euclidean distance measure are used to estimate similarity between the transactions. Three centroid estimation models are tested with two distance measure schemes. Precision and recall and fitness measure are used to test the cluster accuracy levels. Java language and Oracle database are selected for the system development.

Keywords:- Clustering, Data Partitioning, K-means Clustering, Initial Centroid, Inter cluster distance model, optimal Centroid.

1 Introduction

Clustering is the process of organizing data objects into a set of disjoint classes called clusters. Clustering is an example of unsupervised classification. Classification refers to a procedure that assigns data objects to a set of classes. Unsupervised means that clustering does not depends on predefined classes and training examples while classifying the data objects. Cluster analysis seeks to partition a given data set into groups based on specified features so that the data points within a group are more similar to each other than the points in different groups [6], [9]. Therefore, a cluster is a collection of objects that are similar among themselves and dissimilar to the objects belonging to other clusters. Clustering is an crucial area of research, which finds applications in many fields including bioinformatics, pattern recognition, image processing, marketing, data mining, economics, etc.

Cluster analysis is a one of the primary data analysis tool in the data mining. Clustering algorithms are mainly divided into two categories: Hierarchical algorithms and Partition algorithms. A hierarchical clustering algorithm divides the given data set into smaller subsets in hierarchical fashion. A partition clustering algorithm partition the data set into desired number of sets in a single step [9],[17],[20].

Numerous methods have been proposed to solve clustering problem. One of the most popular clustering methods is k-means clustering algorithm developed by Mac Queen in 1967. The easiness of k-means clustering algorithm made this algorithm used in several fields. The k-means clustering algorithm is a partitioning clustering method that separates data into k groups [1], [2], [4], [5], [7], [9]. The k-means clustering algorithm is more prominent since its intelligence to cluster massive data rapidly and efficiently. However, k-means algorithm is highly precarious in initial cluster centers. Because of the initial cluster centers produced arbitrarily, k-means algorithm does not promise to produce the peculiar clustering results[18],[19].

Efficiency of the original k-means algorithm heavily relies on the initial centroids [2], [5]. Initial

centroids also have an influence on the number of iterations required while running the original k-means algorithm. The computational complexity of the original k-means algorithm is very high, specifically for massive data sets [2],[15],[16]. Various methods have been proposed in the literature to enhance the accuracy and efficiency of the k-means clustering algorithm. This paper presents an enhanced method for finding the better initial centroids and to provide an efficient way of assigning the data points to suitable clusters with reduced time complexity.

1.1 The K-Means Algorithm

K-means clustering algorithm is one of the most popular among clustering method. This algorithm generates k points as initial centroids arbitrarily, where k is a user specified parameter. Each point is then assigned to the cluster with the closest centroid [3], [4], [10]. Then the centroid of each cluster is updated by taking the mean of the data points of each cluster. Some data points may move from one cluster to other cluster. Again we calculate new centroids and assign the data points to the suitable clusters. We repeat the assignment and update the centroids, until convergence criteria is met i.e., no point changes clusters, or equivalently, until the centroids remain the same. It is clearly mentioned in the below flowchart. In this algorithm mostly Euclidean distance is used to find distance between data points and centroids [2]. Pseudocode for the k-means clustering algorithm is described in Algorithm 1. The Euclidean distance between two multi-dimensional data points $X = (x_1, x_2, x_3 ... x_m)$ and $Y = (y_1, y_2, y_3 ... y_m)$ is described as follows:

$$d(X,Y)= \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + .. + (x_m - y_m)^2} \quad (1)$$

Algorithm 1: The k-Means Clustering Algorithm
Require: $D = \{d1, d2, d3, ..., d_i, ..., d_n\}$ // Set of n data points.
k // Number of desired clusters
Ensure: A set of *k* clusters.
Steps:
1. Arbitrarily choose *k* data points from *D* as initial centroids;
2. Repeat
Assign each point *d*i to the cluster which has the closest centroid;
Calculate the new mean for each cluster;
Until convergence criteria is met.

Although k-means has the great advantage of being easy to implement, it has some drawbacks. The quality of the final clustering results of the k-means algorithm highly depends on the arbitrary selection of the initial centroids.

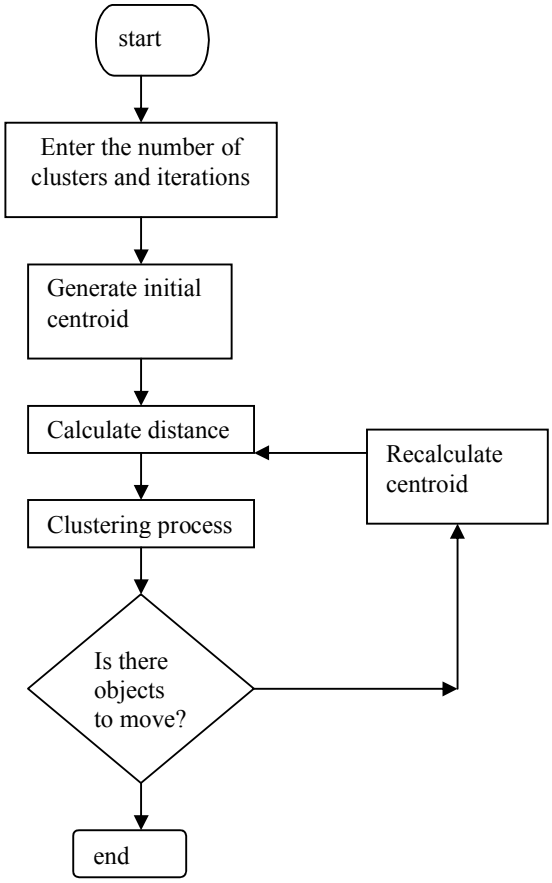

Fig.1 Traditional K-means Algorithm

In the original k-means algorithm, the initial centroids are chosen randomly and hence we get different clusters for different runs for the same input data [10].

Moreover, the k-means algorithm is computationally very expensive also. The computational time complexity of the k-means algorithm is *O(nkl)*, where *n* is the total number of data points in the dataset, *k* is the required number of clusters and *l* is the number of iterations [2]. So, the computational complexity of the k-means algorithm is rely on the number of data elements, number of clusters and number of iterations.

2 Related Work

The original k-means algorithm is very impressionable to the initial starting points. So, it is quite crucial for k-means to have refined initial cluster centers. Several methods have been proposed in the literature for finding the better initial centroids. And some methods were proposed to improve both the accuracy and efficiency of the k-means clustering algorithm. In this paper, some of the more recent proposals are reviewed [1-5], [8].

A. M. Fahim and M. A. Ramadan, [2] proposed an enhanced method for assigning data points to the suitable clusters. In the original K-Means algorithm in each iteration the distance is calculated between each data element to all centroids and the required computational time of this algorithm is depends on the number of data elements, number of clusters and number of iterations, so it is computationally expensive.In Fahim approach the required computational time is reduced when assigning the data elements to the appropriate clusters. But in this method the initial centroids are selected randomly. So this method is very sensitive to the initial starting points and it does not promise to produce the unique clustering results.

K. A. Abdul Nazeer and M. P. Sebastian, [1] Proposed an enhanced algorithm to improve the accuracy and efficiency of the K-Means clustering algorithm. In this algorithm two methods are used, one method for finding the better initial centroids. And another method for an efficient way for assigning data points to appropriate clusters with reduced time complexity. This algorithm produces good clusters in less amount of computational time.

Zhang Chen and Shixiong Xia, [3] proposed the initial centroids algorithm based on k-means that have avoided alternative randomness of initial center. Fang Yuan and R. Dong, [4] proposed the initial centroids algorithm. The standard k-means algorithm selects k-objects randomly from the given data set as the initial centroids. If different initial values are given for the centroids, the accuracy output by the standard k-means algorithm can be affected. In Yuan's method the initial centroids are calculated systematically. Koheri Arai and Ali Ridho Barakbah [5] proposed an algorithm for centroids initialisation for k-means. In this algorithm both k-means and hierarchical algorithms are used. This method utilizes all the clustering results of k-means in certain times. Then, the result transformed by combining with Hierarchical algorithm in order to find the better initial cluster centers for k-means clustering algorithm.

A.Bhattacharya and R. K. De, [8] proposed a novel clustering algorithm, called Divisive Correlation Clustering Algorithm (DCCA) for grouping of genes. DCCA is able to produce clusters, without taking the initial centroids and the value of k, the number of desired clusters as an input. The time complexity of the algorithm is high and the cost for repairing from any misplacement is also high.

Xiaoping Qin et al. [14] proposed an improved algorithm based on the triangle inequality theorem. This algorithm aimed for producing the results which leads to lower time consumption and more effective for large dataset. For better results they carry out the case study in the customer segmentation.

3 Problem Definition

Data clustering techniques are used to partition the transactional data values into a set of groups based on the relevancy. A variety of clustering techniques are used for the data partitioning requirements. Hierarchical and partitioning techniques are widely used clustering methods. Hierarchical clustering considers the data and its structure. The data values are considered in the partitioning clusters. The K-means clustering algorithm is used for the partitioning clustering technique. The K-means clustering algorithm requires the cluster count (K) as the main input data. Centroid values are used to measure transaction relevancy. The user assigns the centroid values in the initial iteration. Different centroid optimization techniques are used. The centroid initialization is done with random transactions. All the clustering results and their accuracy are impacted with reference to the centroid initialization process. Transaction relationship values are estimated using the similarity or distance measure. The proposed system is designed to improve the centroid initialization process.

The proposed system is designed to improve the K-means clustering algorithm with efficient centroid estimation models. Three centroid estimation models are proposed system. They are random selection with distance management, mean distance model and inter cluster distance model. Cosine distance measure and Euclidean distance measure are used to estimate similarity between the transactions. Three centroid estimation models are tested with two distance measure schemes. Precision and recall and fitness measure are used to test the cluster accuracy levels.

3.1 Centroid Optimization Scheme

In this section, we proposed an enhanced method for enhancing the performance of k-means clustering algorithm. In the paper [1] authors proposed an enhanced method to improve the efficiency of the k-means clustering algorithm. But in this method the initial centroids are selected randomly. So this method is very sensitive to the initial starting points and it does not promise to produce the unique clustering results.

In the paper [1] authors proposed an enhanced algorithm to improve the accuracy and efficiency of the k-means clustering algorithm. In this algorithm two methods are used, one method for finding the better initial centroids. And another method for an efficient way for assigning data points to appropriate clusters. In the paper [1] the method used for finding

the initial centroids computationally expensive. In this paper we proposed a new approach for finding the better initial centroids with reduced time complexity. For assigning the data points we follow the paper [1], [2].

The pseudocode for the proposed algorithm is outlined as Algorithm 2.In the proposed algorithm first we are checking, the given data set contain the negative value attributes or not. If the data set contains the negative value attributes then we are transforming the all data points in the data set to the positive space by subtracting the each data point attribute with the minimum attribute value in the given data set. Here, the transformation is required, because in the proposed algorithm we calculate the distance from origin to each data point in the data set. So, for the different data points as showed in Fig. 2, we will get the same Euclidean distance from the origin. This will result in incorrect selection of the initial centroids. To overcome this problem all the data points are transformed to the positive space. Then for all the data points as showed in Fig. 2, we will get the unique distances from origin. If data set contains the all positive value attributes then the transformation is not required.

In the next step, for each data point we calculate the distance from origin. Then, the original data points are sorted accordance with the sorted distances. After sorting partition the sorted data points into k equal sets. In each set take the middle points as the initial centroids. These initial centroids lead to the better unique clustering results. Next, for each data point the distance calculated from all the initial centroids. The next stage is an iterative process which makes use of a heuristic approach to reduce the required computational time. The data points are assigned to the clusters having the closest centroids in the next step. Cluster Id of a data point denotes the cluster to which it belongs. Nearest Dist of a data point denotes the present nearest distance from closest centroid.

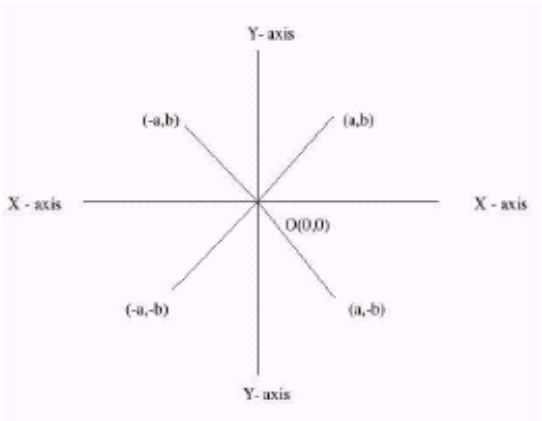

Fig.2 Data Points in Two Dimensional Space

Algorithm 2: The Enhanced Method

Require: $D = \{d_1, d_2, d_3,..., d_i,..., d_n\}$ // Set of n data points.
$d_i = \{ x_1, x_2, x_3,..., x_i,..., x_m \}$ // Set of attributes of one data point. k // Number of desired clusters.
Ensure: A set of k clusters.
Steps:
1: In the given data set D, if the data points contain the both positive and negative attribute values then go to step 2, otherwise go to step 4.
2: Find the minimum attribute value in the given data set D.
3: For each data point attribute, subtract with the minimum attribute value.
4: For each data point calculate the distance from origin.
5: Sort the distances obtained in step 4. Sort the data points accordance with the distances.
6: Partition the sorted data points into k equal sets.
7: In each set, take the middle point as the initial centroid.
8: Compute the distance between each data point d_i $(1 <= i <= n)$ to all the initial centroids c_j $(1 <= j <= k)$.
9: Repeat
10: For each data point d_i, find the closest centroid c_j and assign d_i to cluster$j$
11: Set ClusterId[i]=j. // j:Id of the closest cluster.
12: Set NearestDist[i]= $d(d_i, c_j)$.
13: For each cluster j $(1 <= j <= k)$, recalculate the centroids.
14: For each data point d_i,
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, the data point stays in the same cluster.
Else
14.2.1 For every centroid c_j $(1<=j<=k)$ compute the distance $d(d_i,c_j)$.
End for;
Until the convergence criteria is met.

Next, for each cluster the new centroids are calculated by taking the mean of its data points. Then for each data point the distance calculated from the new centroid of its present nearest cluster. If this distance is less than or equal to the previous nearest distance, then the data point stays in the same cluster, otherwise for each data point we need to calculate the distance from all centroids. After calculated the distances, the data points are assigned to the appropriate clusters and the new Cluster Id's are given and new Nearest Dist values are updated. This reassigning process is repeated until the convergence criterion is met.

4 Proposed Scheme

The proposed system is designed to improve the K-means clustering algorithm with efficient centroid estimation models. Three centroid estimation models are proposed system.

- Random selection with distance management.
- Mean distance model and
- Inter cluster distance model

Transaction relationship is estimated using distance measures. Cosine distance measure and Euclidean distance measure are used to estimate similarity between the transactions. Three centroid estimation models are tested with two distance measure schemes. Precision and recall and fitness measure are used to test the cluster accuracy levels.

The system is divided into four major modules. Centroid estimation module is designed to estimate centroid for clusters. Distance measurement module is designed to calculate transaction distance. Clustering process module is used to partition the transaction datasets. Cluster analysis module is designed to estimate cluster accuracy.

4.1 Implementation of Customer Segmentation

As the customer data has become more and more large, the efficiency is unsatisfactory when using the traditional K-means algorithm. In this section, a practical dataset from a mobile communication company is employed to carry out customer segmentation. Based on the research for the customer segmentation, three main attributes of customer dataset are adopted as essential standards i.e., R-Recency, F- Frequency and M-Monetary. With this three factors, First establish model called RFM model.

We employ the weights of RFM:[„ *F R M W W W*] =[0.221, 0.341, 0.439][13]. We can see the weight of M is the highest. Experts believe that the payment fee amount of customer is the most important attribute. Then we can divide customers into eight groups, that is to say, the number of clusters is eight.

4.1.1 Data Structure of K-Means New Algorithm

The data structure to implement the improved algorithm is as follows:

(1) Record the information of a cluster Public Structure cluster

Double Mean	'Mean of cluster
Int ClusterNum	'Number of clusters
Int CluserId	'ID of clusters
Double SquareError	'Square error of clusters

End Structure

(2) record the information of a sample point in a cluster

Public Structure sample point

Int CluserId	'ID of clusters
ArrayList recordid	'Recording of key words

End Structure

(3) Record the distance between each cluster center double dist [100] 'Distance between each cluster center.

In order to implement the improved algorithm, we employed some important functions as follows:

(1) InitClusters ( ): Initial cluster centers

(2) Calc_meandist(): Calculate distance between each cluster center

(3) Calc_samdist(): Calculate distance between each item to each cluster center.

(4) Calc_dist( ): Calculate distance between two items

(5) FindClosestCluster( ): Find the cluster whose centroid is nearest and assign the item to the corresponding cluster

(6) CalcNewClustCenters(): Calculate new cluster centers

(7) CalcErrorSum( ): Calculate sum of squared error

(8) RunK-Means( ): Run K-Means algorithm

(9) ShowClusters( ): Display customer data of each cluster

In the process of clustering, users are required to input allowable error criteria and the number of clusters.

4.1.2 Improved K-Means Algorithm

In the above K-Means algorithm, we assign each item to the cluster whose centroid is nearest (distance is computed by using Euclidean distance [4],[22],with either standardized or un-standardized observations). The time complexity of this progress is O(*nkd*). *n* refers to the number of the total items, and *k* refers to the number of clusters initially set, and *d* refers to the vector dimension of data objects. We need to recalculate the centroid for the cluster receiving the new item or for the cluster losing the item, and the time complexity of this progress is O(*nd*). Consequently, the total time complexity of one iteration is O(*nkd*). The algorithm's time consumption is fairly considerable when the dataset is large.

When we use traditional K-Means algorithm, in the first iteration stage, we calculate the distance from each item to every cluster center, then compare these distances and calculated the cluster whose centroid is nearest, and then the item is assigned to the cluster. If we can avoid unnecessary comparison and distance calculation through some way, the time consumption will be reduced. As the distance is computed by using Euclidean distance, we can consider employing the geometry theorem of triangle trilateral relationship:

the sum of two sides is greater than the third side in a triangle. In this way, we can simplify the calculation. Let $x_l \in X$, $d(c_n,c_m)$ is the distance between two cluster centers, $d(c_n,c_m)$, $d(x_l,c_n)$ and $d(x_l,c_m)$form a triangle as shown in fig. 3. According to the geometry theorem of triangle trilateral relationship, there is a equation as follows:

$$d(c_n,c_m) \leq d(x_l,c_n)+d(x_l,c_n) \quad (2)$$

Then

$$d(c_n,c_m)- d(x_l,c_n) \leq + d(x_l,c_m) \quad (3)$$

If $d(c_n,c_m) \geq 2d(x_l,c_n)$, then: $d(x_l,c_n) \leq d(x_l,c_m)$

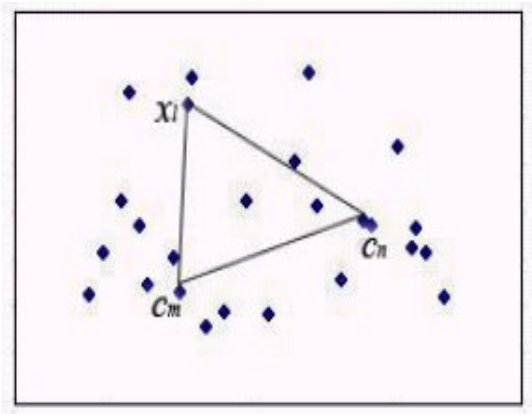

Fig. 3 Geometry Theorem of Triangle Trilateral Relationship

Thus, it is unnecessary to calculate $d(x_l,c_m)$ on condition that $d(c_n,c_m) \geq 2d(x_l,c_n)$. Thus we can improve K-Means based on theory above, and we call the improved algorithm K-Means_new.Now let's make a comparison between K-Means and K-Means_new. When we re-calculate cluster centers in the second iteration stage, the time complexity of two algorithms is equal. However, when we calculate clusters in the first iteration stage, the computational complexity of K-Means_new algorithm is reduced. First, we consider one sample point. When we use traditional K-Means algorithm to calculate the distance from each item to every cluster center, the number of calculation is *k*. However, when we use K-Means_new, the number of calculation is 1 in the best situation and that is *k* in the worst case. We suppose that α is the average number of calculation in the first iteration stage, then: $\alpha \prec k$. Therefore, when we use K-Means, the time complexity of one iteration is O(*nkd*) and that is O(*n* α*d*) when we use K-Means_new. If dataset is comparatively large, that is to say *n* is large, the superiority of the improved algorithm will turn up.

5 Experimental Results and Analysis

By cluster analysis, we can get eight classes of customers. The specific experimental results are shown in Table 1.

Table 1 Results of Cluster Analysis

No.	size	R	F	M	Customer category
1	364	45.82	6.57	879.36	Quite important customer
2	296	32.24	4.49	689.23	Developing customer
3	123	59.36	9.58	769.43	Quite Important customer
4	312	48.45	7.24	453.52	Important customer
5	345	84.77	6.68	644.69	Common customer
6	301	70.39	4.45	442.75	Worthless customer
7	135	75.64	6.80	1011.32	Retent customer
8	352	66.52	4.06	799.89	Common customer
Mean		60.40	6.23	711.27	

As can be seen in Table 1, customers of company are clustered into eight classes. Then telecom enterprise can adopt different management strategies for different classes of customers.

Table 2 Accuracy Level Analysis

Samples	**K-Means**	**K-Means-C**	**K-Means-OC**
50	63	82	90
100	65	83	92
150	66	85	93
200	68	87	95
250	70	88	96

The accuracy level is analyzed in the system (Table 2). Fitness functions are used in the accuracy level analysis. K-means algorithm, K-means with improved initial centroids and K-means with optimized centroids schemes are used in the analysis. The K-means clustering with improved initial centroids scheme produces 20% better then K-means clustering algorithm. K-means clustering with centroid optimization scheme produces 30% better than K-means clustering algorithm.

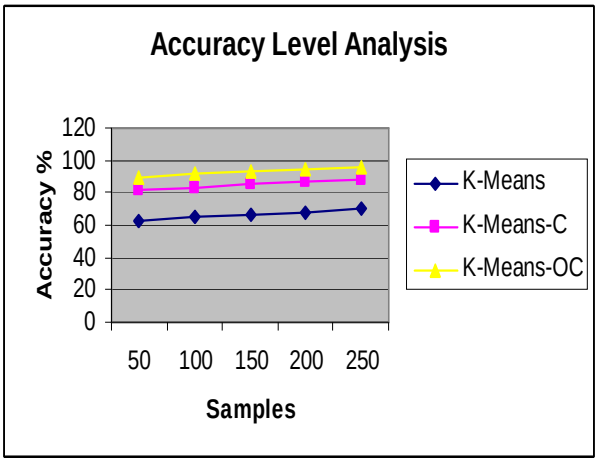

Fig. 4 Accuracy Level Analysis

5.1 Analysis Methods

There are many methods exists like F-measure, purity, entropy and separation index to evaluate the accuracy of the clustering algorithms[21]. Speed up measure is also used to evaluate the communication criteria for the peer nodes. Network size and height parameters are also analyzed in the system. Some of the metrics here used for calculating the precision and recall value in order to find the accuracy of the cluster. Separation index value will give the distance between the two clusters so that the two clusters are distinct which will give the high level of accuracy in the clustering process.

5.1.1 F-measure

The F-measure is a harmonic combination of the precision and recall values used in information retrieval. Each cluster obtained can be considered as the result of a query, whereas each pre classified set of documents can be considered as the desired set of documents for that query. Thus, we can calculate the precision P(i, j) and recall R(I, j) of each cluster j for each class i.

If n_i is the number of members of the class i, n_j is the number of members of the cluster j, and n_{ij} is the number of members of the class i in the cluster j, then P(i, j) and R(i, j) can be defined as

$$P(i,j) = \frac{n_{ij}}{n_j}, \tag{4}$$

$$R(i,j) = \frac{n_{ij}}{n_i} \tag{5}$$

The corresponding F-measure F(i, j) is defined as

$$F(i,j) = \frac{2 * P(i,j) * R(i,j)}{P(i,j) + R(i,j)} \tag{6}$$

Then, the F-measure of the whole clustering result is defined as

$$F = \sum_i \frac{n_i}{n} \max_j (F(i,j)), \tag{7}$$

where n is the total number of documents in the data set. In general, the larger the F-measure is, the better the clustering result is [2].

5.1.2 Purity

The purity of a cluster represents the fraction of the cluster corresponding to the largest class of documents assigned to that cluster; thus, the purity of the cluster j is defined as

$$Purity(j) = \frac{1}{n_j} \max_i (n_{ij}) \tag{8}$$

The overall purity of the clustering result is a weighted sum of the purity values of the clusters as follows:

$$Purity = \sum_j \frac{n_j}{n} Purity(j) \tag{9}$$

In general, the larger the purity value is, the better the clustering result.

5.1.3 Separation Index

SI is another cluster validity measure that utilizes cluster centroids to measure the distance between clusters, as well as between points in a cluster to their respective cluster centroid. It is defined as the ratio of

average within-cluster variance (cluster scatter) to the square of the minimum pairwise distance between clusters:

$$\mathrm{SI} = \frac{\sum_{i=1}^{Nc}\sum_{x_j \in c_i} dist(x_j, m_i)2}{N_D \min_{1\le r,r\le Nc, r\neq\infty}\{dist(m_r, m_s)\}^2} \quad (10)$$

$$= \frac{\sum_{i=1}^{Nc}\sum_{x_j \in c_i} dist(x_j, m_i)^2}{N_D \cdot dist_{\min}^2} \quad (11)$$

where m_i is the centroid of cluster c_i, and distmin is the minimum pairwise distance between cluster centroids. Clustering solutions with more compact clusters and larger separation have lower Separation Index, thus lower values indicate better solutions. This index is more computationally efficient than other validity indices, such as Dunn's index [11],[12] which is also used to validate clusters that are compact and well separated. In addition, it is less sensitive to noisy data.

6 Conclusion

The most popular clustering algorithms is k-means clustering algorithm which is a partitioning technique The quality of the final clusters rely heavily on the initial centroids, which are selected randomly in the traditional K-means algorithm. Moreover, the k-means algorithm is computationally very expensive also. The proposed algorithm is found to be more accurate and efficient compared to the original k-means algorithm.

This proposed method finding the better initial centroids by using the optimal centroid estimation process and provides an efficient way of assigning the data points to the suitable clusters. In this method the inter cluster distance is considered which gives an accurate result. This method ensures the total mechanism of clustering in O(n log n) time without loss the correctness of clusters. This approach does not require any additional inputs like threshold values. The proposed algorithm produces the more accurate unique clustering results. The value of k, desired number of clusters is still required to be given as an input to the proposed algorithm. Automating the determination of the value of k is suggested as a future work.

References:

[1] K. A. Abdul Nazeer and M. P. Sebastian, "Improving the accuracy and efficiency of the k-means clustering algorithm," in International Conference on Data Mining and Knowledge Engineering (ICDMKE) Proceedings of the World Congress on Engineering (WCE-2009), Vol 1, July 2009, UK.

[2] A. M. Fahim, A. M. Salem, F. A. Torkey and M. A. Ramadan, "An Efficient enhanced k-means clustering algorithm," journal of Zhejiang University, 10(7): 16261633, 2006.

[3] Chen Zhang and Shixiong Xia, " K-means Clustering Algorithm with Improved Initial center," in Second International Workshop on Knowledge Discovery and Data Mining (WKDD), pp. 790-792, 2009.

[4] F. Yuan, Z. H. Meng, R. Dong, "A New Algorithm to Get the Initial Centroids," proceedings of the 3rd International Conference on Machine Learning and Cybernetics, pp. 26-29, August 2004.

[5] Koheri Arai and Ali Ridho Barakbah, "Hierarchical K-means: an algorithm for Centroids initialization for k-means," department of information science and Electrical Engineering Politechnique in Surabaya, Faculty of Science and Engineering, Saga University Vol. 36, No.1, 2007.

[6] S. Deelers and S. Auwatanamongkol, "Enhancing K-Means Algorithm with Initial Cluster Centers Derived from Data Partitioning along the Data Axis with the Highest Variance," International Journal of Computer Science, Vol. 2, Number 4.

[7] Mc Queen J, "Some methods for classification and analysis of multivariate observations," Proc. 5th Berkeley Symp. Math. Statist. Prob., (1): 281–297, 1967.

[8] A. Bhattacharya and R. K. De, "Divisive Correlation Clustering Algorithm (DCCA) for grouping of genes: detecting varying patterns in expression profiles," bioinformatics, Vol. 24, pp. 1359-1366, 2008.

[9] Margaret H Dunham, Data Mining-Introductory and Advanced Concepts, Pearson Education, 2006.

[10] Elmasri, Navathe, Somayajulu, Gupta, Fundamentals of Database Systems, Pearson Education, First edition, 2006.

[11] (2010) The UCI Repository website. [Online]. Available: http://archive.ics.uci.edu/

[12] Height-Weight Data. (2010). [Online]. Available:http://www.disabledworld.com/artman/publish/height-weight-teens.shtml

[13] Diggle Data. (2010). Available: http://lib.stat.cmu.edu/datasets/diggle

[14] Xiaoping Qin, Shijue Zheng, Ying Huang and Guangsheng Deng "Improved K-Means algorithm and application in customer segmentation" Asia-Pacific Conference on Wearable Computing Systems, 2010.

[15] Madhu Yedla, Srinivasa Rao and T M Srinivasa "Enhancing K-means Clustering Algorithm with Improved Initial Center" Madhu Yedla et al. / International Journal of Computer Science and Information Technologies, Vol. 1, 2010, 121-125.

[16] Su M.-C. and C.-H. Chou, "A Modified Version of the k-Means Algorithm with a Distance Based on Cluster Symmetry," *IEEE Trans. Pattern Analysis and Machine Intelligence*,vol.23,no.6,pp. 674-680, June 2001.

[17] Lockwood J. W, S. G. Eick, D. J. Weishar,R P.Loui, J. Moscola, C. Kastner, A. Levine, and M. Attig, " Transformation Algorithms for Data Streams, 2005 *IEEE Aerospace Conference*", March 5-12, 2005.

[18] Jensen C, D. Lin, and B.C. Ooi, "Continuous Clustering of Moving Objects, " IEEE *Transactions on Knowledge and Data Engineering*, vol. 19, no. 9, pp. 1161-1173, Sept.2007.

[19] Hall, L.O., Ozyurt, B., and Bezdek, J.C. 1999.Clustering with a genetically optimized approach. *IEEE Trans. on Evolutionary - Computation*, 3, 2,103-112.

[20] Guha S, A. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan, "Clustering datastreams:Theory and practice," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 3,pp. 515–528, 2003.

[21] Aggarwal C.C and P.S. Yu, "Redefining Clustering for High Dimensional Applications," IEEE *Transactions on Knowledge and Data Engineering*, vol. 14,no.2, pp. 210-225, Mar./Apr. 2002.

[22] Abdun Naser Mahmood, Christopher Leckie, and Parampalli Udaya "An Efficient Clustering Scheme to Exploit Hierarchical Data in Network Traffic Analysis ". In *IEEE Transactions on Knowledge and Data Engineering*, Vol 20, No 6, June 2008.

About the Authors

M. Vijayakumar has completed his Bachelor of Engineering in Computer Science in Bharathiar University, Tamilnadu, India and Master of Engineering , in Computer Science in Anna University,Chennai, Tamilnadu, India. He has started his teaching profession in the year 2004 in Muthayammal Engineering College, Tamilnadu. India., At present, he is working as an Assistant Professor in the department of Computer Science and Engineering in Sasurie College of Engineering, Tamilnadu. India. He has published 15 research papers in National and International journals and conferences. Currently he is a part time Research Scholar in Anna university of Technology, Coimbatore. His areas of interest are Data mining, Knowledge Engineering, Clustering algorithms and Network security. He has completed 8 years of teaching service.He is a life member of ISTE.

S. Prakash has completed his M.E Computer Science and Engineering in K.S.Rangasamy College of Technology, Tamilnadu, India in 2006. He is a part time research scholar in Anna University of Technology, Coimbatore. He has published 12 research papers in National and International journals and conferences. His areas of interest are Data mining, Knowledge Engineering and Association Rule Mining. Currently, he is working as Assistant Professor in the department of Information Technology, Sasurie College of Engineering, Tamilnadu. India. He has completed 9 years of teaching service. He is a life member of ISTE.

Dr. R.M.S. Parvathi has completed her Ph.D., degree in Computer Science and Engineering in 2005 in Bharathiar University, Tamilnadu, India. Currently she is a Principal and Professor , Department of Computer Science and Engineering in Sengunthar College of Engineering, Tiruchengode, Tamilnadu, India, She has completed 21 years of teaching service in various reputed Institutions. She has published more than 28 articles in International/ National Journals.She has authorized 3 books with reputed publishers. She is guiding 20 Research scholars. Her research areas of interest are Software Engineering, Data Mining, Knowledge Engineering, and Object Oriented System Design. She is a life member of ISTE and Indian Computer Society.