Clustering the Mixed Numerical and Categorical Dataset using Similarity Weight and Filter Method

M. V. Jagannatha Reddy[1] and B. Kavitha[2]

[1]Department of Computer Science, Rayalaseema University, Kurnool-518007
[2]Department of MCA, Sree Vidyanikethan Engineering College, A. Rangampet, Chittoor (dt). A.P. India-51710
mvjagannathreddy@yahoo.co.in, ballikavitha@yahoo.co.in

Abstract

Clustering is a challenging task in data mining technique. The aim of clustering is to group the similar data into number of clusters. Various clustering algorithms have been developed to group data into clusters. However, these clustering algorithms work effectively either on pure numeric data or on pure categorical data, most of them perform poorly on mixed categorical and numerical data types in previous k-means algorithm was used but it is not accurate for large datasets. In this paper we cluster the mixed numeric and categorical data set in efficient manner. In this paper we present a clustering algorithm based on similarity weight and filter method paradigm that works well for data with mixed numeric and categorical features. We propose a modified description of cluster center to overcome the numeric data only limitation and provide a better characterization of clusters. The performance of this algorithm has been studied on benchmark data sets.

Keywords — Data Mining, Clustering, Numerical Data, Categorical Data, K-Prototype, Similarity Weight, Filter Method

1. Introduction

Clustering is a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters. It is challenging task in data mining techniques [7]. Many data mining applications require partitioning of data into homogeneous clusters from which interesting groups may be discovered, such as a group of motor insurance policy holders with a high average claim cost, or a group of clients in a banking database showing a heavy investment in real estate. To perform such analyses at least the following two problems have to be solved; (1) efficient partitioning of a large data set into homogeneous groups or clusters, and (2) effective interpretation of clusters. This paper proposes a solution to the first problem and suggests a solution to the second. With the amazing progress of both computer hardware and software, a vast amount of data is generated and collected daily. There is no doubt that data are meaningful only when one can extract the hidden information inside them. However, "the major barrier for obtaining high quality knowledge from data is due to the limitations of the data itself". These major barriers of collected data come from their growing size and versatile domains. Thus, data mining that is to discover interesting patterns from large amounts of data within limited sources (i.e., computer memory and execution time) has become popular in recent years. Clustering is considered an important tool for data mining. The goal of data clustering is aimed at dividing the data set into several groups such that

objects have a high degree of similarity to each other in the same group and have a high degree of dissimilarity to the ones in different groups. Each formed group is called a cluster. Useful patterns may be extracted by analyzing each cluster. For example, grouping customers with similar characteristics based on their purchasing behaviors in transaction data may find their previously unknown patterns. The extracted information is helpful for decision making in marketing.

Various clustering applications have emerged in diverse domains. However, most of the traditional clustering algorithms are designed to focus either on numeric data [13] or on categorical data [4, 8~11]. The collected data in real world often contain both numeric and categorical attributes. It is difficult for applying traditional clustering algorithm directly into these kinds of data. Typically, when people need to apply traditional distance-based clustering algorithms [3, 18] (ex., k-means [1]) to group these types of data, a numeric value will be assigned to each category in this attributes. Some categorical values, for example "low", "medium" and "high", can easily be transferred into numeric values. But if categorical attributes contain the values like "red", "white" and "blue" … etc., it cannot be ordered naturally. How to assign numeric value to these kinds of categorical attributes will be a challenge work.

In this paper we first divide [2] the original data set into pure numerical and categorical data set. Next, existing well established clustering algorithms [8, 13, 19] designed for different types of datasets are employed to produce corresponding clusters. Last, the clustering results on the categorical and numeric dataset are combined as a categorical dataset on which the categorical data algorithm [14] is employed to get the final output.

The reminder of this paper is organized as follows. Next we show the background and related works and the proposed method for clustering on mixed categorical and numerical data, finally the conclusion of our work.

2. Related Work

2.1 Cluster Ensemble Approach for Mixed Data

Dataset with mixed data type are common in real life. Cluster Ensemble [3] is a method to combine several runs of different clustering algorithm to get a common partition of the original dataset. In the paper divide and conquers technique [2] is formulated. Existing algorithm use similarity measures like Euclidean distance [12] which gives good result for the numeric attribute. This will not work well for categorical attribute. In the cluster ensemble approach numeric data [13] are handled separately and categorical data are handled separately. Then both the results are then treated in a categorical manner. Different types of algorithm used for categorical data are K-Modes [8, 19], K-Prototype, ROCK [4] and squeezer algorithm. In K-Mode the total mismatch of categorical attributes of two data record is projected. The Squeezer algorithm yields good clustering result, good scalability and it handles high dimensional data set efficiently.

2.2 Methodology

1. Splitting of the given data set into two parts. One for numerical data and another for categorical data
2. Applying any one of the existing clustering algorithms for numerical data set
3. Applying any one of the existing clustering algorithms for categorical data set

4. Combining the output of step 2 and step 3

5. Clustering the results using squeezer algorithm

The dynamic dataset are used and measures the cluster accuracy and cluster error rate. The cluster accuracy 'r'is defined by

$$r = \frac{\sum_{i=1}^{k} a_i}{n}$$

Where K represents number of clusters a_i represents number of instance occurring in both the cluster I and its class and n represents number of instance in the dataset. Finally cluster error rate 'e' defined by

$$e = 1 - r$$

Where r represents cluster accuracy.
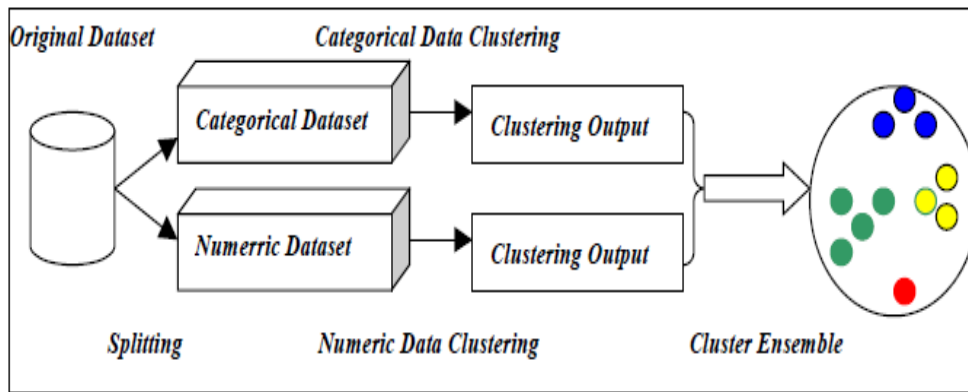
Figure 1. Overview of Cluster Ensemble Algorithm Framework

Figure 1 shows that original dataset has to be splitted into categorical dataset and numerical dataset and clustering them. The output of clustering datasets are clustering using cluster ensemble algorithm. The algorithm is compared with k-prototype algorithm. Fig 2. Shows error rate among the k-prototype and Cluster Ensemble algorithm.
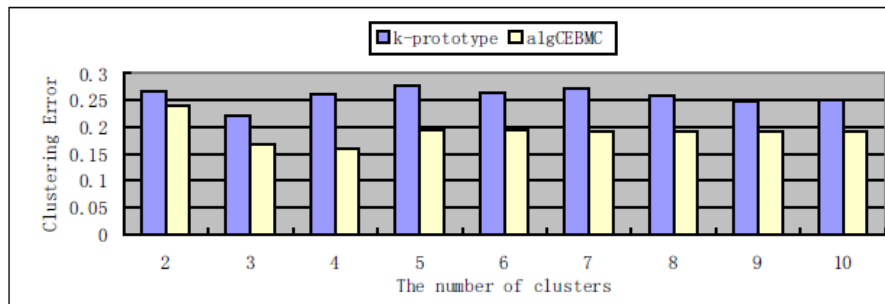
Figure 2. Comparing K-prototype and CEBMC

3. Review of K-means Algorithm

3.1 K-Means

K-means [1] is a clustering algorithm that deals with categorical data only. The k-means clustering algorithm [1] requires the user to specify from the beginning the number of clusters to be produced and the algorithm builds and refines the specified number of clusters. Each cluster has a mode associated with it. Assuming that the objects in the data set are described by m categorical attributes, the mode of a cluster is a vector $Q=\{q1, q2,…, qm\}$ where qi is the most frequent value for the ith attribute in the cluster of objects.

Given a data set and the number of clusters k, the k-means algorithm clusters the set as follows:

1. Select initial k means for k clusters.

2. For each object X

    a. Calculate the similarity between object X and the means of all clusters.

    b. Insert object X into the cluster c whose mode is the most similar to object X.

    c. Update the mode of cluster c

3. Retest the similarity of objects against the current modes. If an object is found to be closer to the mode of another cluster rather than its own cluster, reallocate the object to that cluster and update the means of both clusters.

4. Repeat 3 until no or few objects change clusters after a full cycle test of all the objects.

In most clustering algorithms, an object is usually viewed as a point in a multidimensional space. It can be represented as a vector ($x1...xd$), a collection of values of selected attributes with d dimensions; and xi is the value of i-th selected attribute. The value of xi may be numerical or categorical.

Most pioneers of solving mixed numeric and categorical value for clustering problem is to redefine the distance measure and apply it to existing clustering algorithms.

The algorithm consists of a simple re-estimation procedure as follows. Initially, the data points are assigned at random to the K sets. For step 1, the centroid is computed for each set. In step 2, every point is assigned to the cluster whose centroid is closest to that point. These two steps are alternated until a stopping criterion is met, i.e., when there is no further change in the assignment of the data points.

3.2 K-Prototype

K-prototype is one of the most famous methods. K-prototype inherits the ideas of k-means [1], it applies Euclidean distance [12] to numeric attributes and a distance function is defined to be added into the measure of the closeness between two objects. Object pairs with different categorical values will enlarge the distance between them. The main shortcomings of k-prototype may fall into followings:

(1) Binary distance is employed for categorical value. If object pairs with the same categorical value, the distance between them is zero; otherwise it will be one. However, it will not properly show the real situation, since categorical values may

have some degree of difference. For example, the difference between "high" and "low" shall not equal to the one between "high" and "medium".

(2) Only one attribute value is chosen to represent whole attribute in cluster center. Therefore, the categorical value with less appearance seldom gets the chance to be shown in cluster center, though these items may play an important role during clustering process. Additionally, since k-prototype inherits the ideas of k-means, it will retain the same weakness of k-means.

4. Proposed Algorithm

In this paper we propose a new algorithm for clustering mixed numerical and categorical data. In this algorithm we do the following: First we read the large data set D.

Split [2] the Data Set D into Numerical Data and Categorical Data. Store all the Data Set. Cluster the Numerical data set and categorical using Similarity Weight.

$$Sim(a,b) = \sum_{i=1}^{n} t\sqrt{(a_i - b_i)^2}$$

Combine [2] the clustered categorical dataset and clustered numerical dataset as a categorical dataset using Filtered method.

In this algorithm we cluster the numerical data, categorical data and mixed data.

4.1 Similarity Weight Method

Measuring similarity between two data objects is one of the challenging tasks in data mining. To measure the 'similarity' of two sets of clusters, we define a simple formula here: Let C={ $C_1, C_2, C_3, \ldots, C_m$ } and D={ $D_1, D_2, \ldots, D_n$ } be the results of two clustering algorithms on the same data set. Assume C and D are "hard" or exclusive clustering algorithms where clusters produced are pair-wise disjoint, i.e., each pattern from the dataset belongs to exactly one cluster. Then the similarity matrix for C and D is an m × n matrix $.S_{C,D}(1)$.

$$S_{c,d} = \begin{bmatrix} S_{11} & S_{12} \ldots S_{1j} \ldots S_{1n} \\ \ldots \ldots \\ S_{i1} & S_{i2} \ldots S_{ij} \ldots S_{in} \\ \ldots \ldots \\ S_{m1} & S_{m2} \ldots S_{mj} \ldots S_{mn} \end{bmatrix}$$

where Sij = p/q, which is Jaccard's Similarity Coefficient [7] with p being the size of intersection and q being the size of the union of cluster sets Ci and Dj. The similarity of clustering C and clustering D is then defined as

$$Sim(C, D) = \sum_{i \leq m, j \leq n} S_{ij} / \max(m, n)$$

For Example 1, let C_1={1,2,3,4} C_2 ={5,6,7,8} and D_1={1,2}, D_2 ={3,4}, D_3 ={5,6}, D_4={7,8} thus m=4 and n=2, then the similarity between clustering C and D is given by the following matrix D_1.

Table 1. Similarity Matrix on Example1 Data

Cluster	D_1	D_2	D_3	D_4
C_1	2/4	2/4	0/6	0/6
C_2	0/6	0/6	2/4	2/4

In cell $C_1 D_1$, p=$|C_1 \cap D_1|$=|{1,2}|=2, and q=$|C_1 U D_1|$=|{1,2,3,4}|=4. Therefore, cell $C_1 D_1$=p/q=2/4.

Similarly the other cells of the matrix are calculated. Thus, the similarity between cluster set C and cluster set D in this case is Sim(C, D) = (2/4+2/4+0/6+0/6+0/6+0/6+2/4+2/4)/4 = 0.5

Table 2. Similarity Matrix on Example1 Data

Cluster	D_1	D_2
C_1	4/6	2/8
C_2	0/6	2/4

For Example 2, let C_1 ={1,2,3,4,5,6} C_2 ={7,8}; D_1 ={1,2,3,4}, D_2 ={5,6,7,8}, thus m=2, n=2 and matrix $S_{C,D}$ is:

Thus, the similarity Sim(C, D), according to the similarity matrix above, is (4/6+2/8+0/6+2/4)/2= 17/24 or 0.7083 It is easy to show that 0 < Sim(C, D) ≤ 1; and Sim(C, D)=1 for two identical clustering, where the similarity matrix Sim(C, D) is a square matrix; and that this measure is only applicable to clustering a finite set of patterns into a finite number of disjoint (or non-overlapping) clusters. Also, we can take the square of summation of the matrix values to define similarity Sim(C, D), i.e., let **Sim(C, D) = (Σi,j Sij / max(m, n))2**, this would have the effect of giving a lower value of similarity but without changing its range of (0, 1]. This similarity measure is a reasonable one to use because, if we define the dissimilarity or "distance" between two clusterings C and D as U (C,D) = 1 – Sim (C,D), then it can be proved that U(C,D) is a good distance measure for it satisfies all desirable properties (non-negativity, identity, symmetry, triangle inequality) of a distance metric.

4.2 Clustering Similarity Analysis

After applying clustering algorithms on datasets, cluster similarities were calculated using similarity weight method. The results were then verified by calculating centroid Euclidean distance and Pearson correlation.

Euclidean distance:

$$d = \sqrt{\sum (X - Y)^2}$$

Pearson correlation coefficient:

$$r = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{\left[\sum X^2 - \frac{(\sum X)^2}{N}\right]\left[\sum Y^2 - \frac{(\sum Y)^2}{N}\right]}}$$
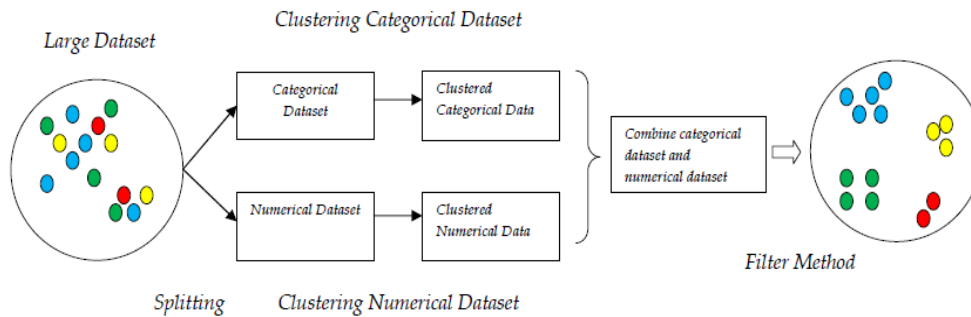
Figure 3. Overview of Similarity and Filtered Algorithm Framework

4.3. Filter Algorithm

For clustering the mixed numerical and categorical datasets, we proposed an algorithm called filter method. First, the original dataset is divided [2] into two sub-datasets i.e., pure categorical dataset and the pure numerical dataset. Next, we apply clustering algorithms on sub-datasets [8, 13] based upon their type of dataset to get the corresponding clusters. Last, the clustering results of numerical and categorical datasets are combined as a categorical dataset, on which the categorical data clustering algorithm is exploited to get the final cluster. Fig.3 shows the overview of similarity and filtered algorithm framework.

Now we discuss the last step of above process. With the clustering results on the categorical and numerical datasets, we also get the final clustering results by exploiting filter method.

The process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints and data sources, etc. Applications of collaborative filtering typically involve very large data sets. Collaborative filtering methods have been applied to many different kinds of data including sensing and monitoring data - such as in mineral exploration, environmental sensing over large areas or multiple sensors; financial data - such as financial service institutions that integrate many financial sources; or in electronic commerce and web 2.0 applications where the focus is on user data, etc. The remainder of this discussion focuses on collaborative filtering for user data, although some of the methods and approaches may apply to the other major applications as well.

Step(1):   Read the dataset D

Step(2):   Compute Similarity Matrix

Step(3):   Compute F

Step(4):   Cluster data according to F

Step(5):   Submit F to Filter method

4.3.1 Filter Algorithm

Input: Categorical Data set DS

Output: Set of Clusters

$$D=\{X_1,X_2,….X_n\}$$

$$C = \text{No of Clusters}$$

$$m= \text{No of Attributes}$$

$$n= \text{No of Records}$$

$$F(D,C) = \sum_{t=1}^{m}\sum_{i=1}^{n} w_{i,t} d(X_i,C_t)$$
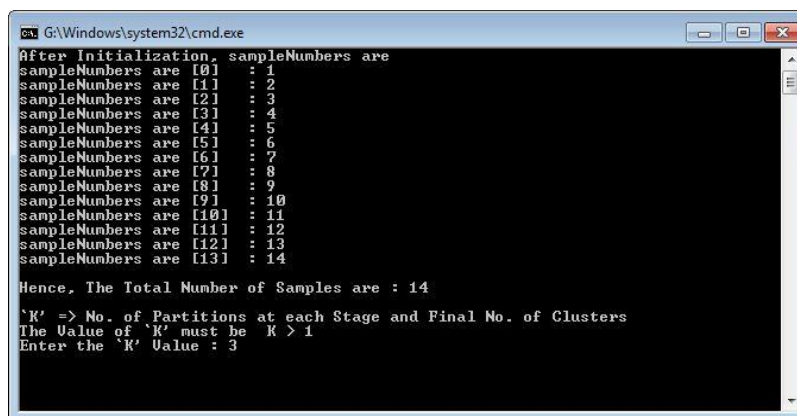
1. Read Data set D

2. Compute Dissimilarity

$$d(x,y) = \frac{x^2 \cdot y^2}{|x| \cdot |y|}$$

3. Compute F

$$F(D,C) = \sum_{t=1}^{m}\sum_{i=1}^{n} w_{i,t} d(X_i,C_t)$$

4. Cluster data according to F.

5. Submit the F into post Clustering Technique

   i.e. Filtered Algorithm

Figure 4. Initialization and Input

Figure 5. Applying Clustering Technique Similarity Weight and Filter Method

Figure 6. Results of Clustering Showing Groups Divided into Clusters

Figure 7. Final Clustering Results

4.3 Advantages of Proposed System

- ✓ Efficient use of the cut and cycle properties by our Fast Filtered-Inspired clustering algorithm.

- ✓ Shape of a cluster has very little impact on the performance of this Filtered clustering algorithm.

- ✓ Efficient for dimensionality more than 5 and reduced time complexity.

- ✓ Nearest neighbor search is used to construct efficient Filtered.

- ✓ Works efficiently even if the boundaries of clusters are irregular.

5. Clustering results

We used the filter and k-prototypes algorithms to cluster the dynamic dataset into different number of clusters, varying from 1 to 10. For each fixed number of clusters, the clustering errors of different algorithms were compared. Screen shots are shown in Figures 4, 5, 6 and 7.

For both of datasets, the k-prototypes algorithm, just as has been done. All numeric attributes are rescaled to the range of [0.1].

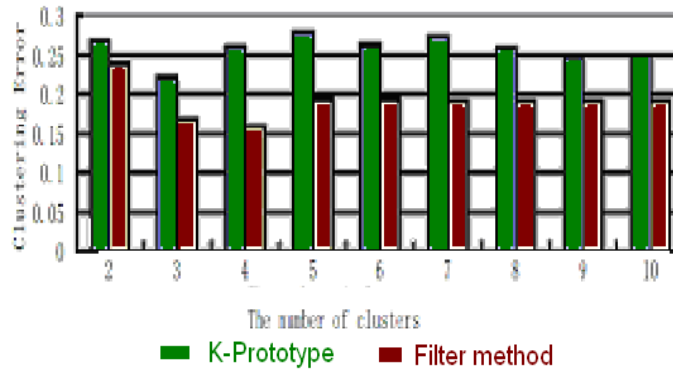**Figure 8. Clustering Error vs. Number of Clusters**

Figure 8 shows the results on the dynamic dataset of different clustering algorithms from Figure 7, we can summaries the relative performance of our algorithms as follows.

Table 3. Relative Performance of Different Clustering Algorithms

Method	Average clustering error
K-Prototype	0.311
Similarity & Filter	0.181

That is, comparing with the k-prototypes algorithm, our algorithm performed the best in all cases. It never performed the worst. Furthermore, the average clustering errors of our algorithm are significantly smaller than that of the k-prototypes algorithm. The above experimental results demonstrate the effectiveness of filter method for clustering dataset with mixed attributes. In addition, it outperforms the k-prototypes algorithm with respect to clustering accuracy.
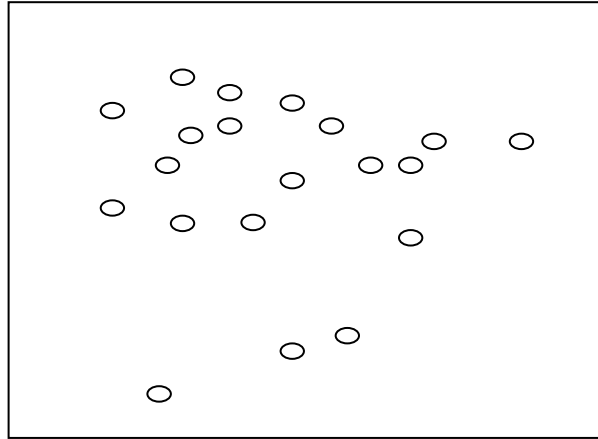

Fig 9: Final Clusters of K-Prototype

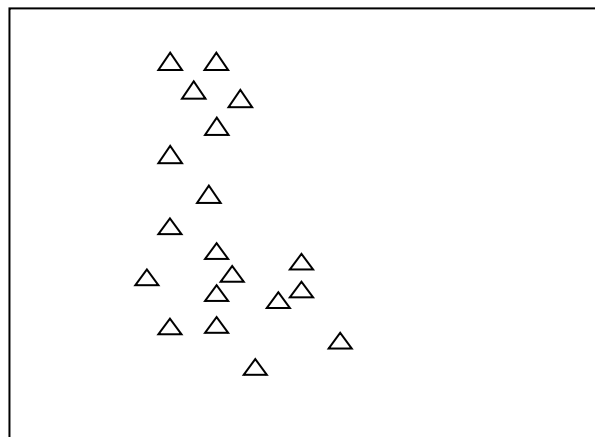Figure 9 shows the final cluster of mixed numeric and categorical datasets using K-Prototype.

Fig 10: Final Clusters of Filter Method

Figure 10 shows the final cluster of mixed numeric and categorical datasets using Similarity Weight and Filter Method.
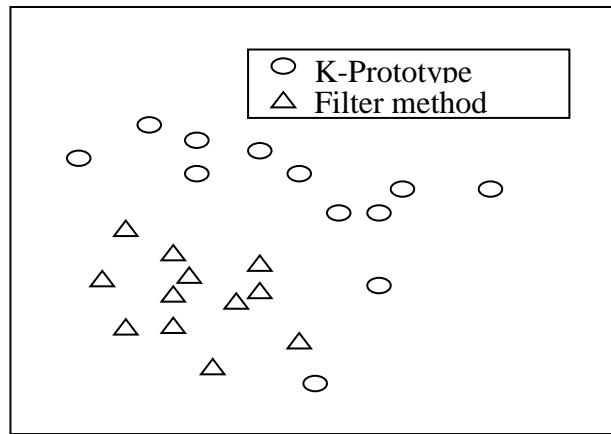
Fig 11: Locations of Solutions

Figure 11 shows the locations where the mixed numeric and categorical data set are clustered. From the figure we conclude that filter method gives better solution than the K-Prototype algorithm.

6. Conclusion

We concluded that filter method is showing good results than the K-Prototype algorithm. The method of comprehensive assessment using similarity weight in the attribute synthetic evaluation system seemed to be objective and rational. Not only it embodied the weights of variables involved, but also exploiting the information presented by the sample. Our system is efficient for any number of Dimensions and reduces Time Complexity. Also Irregular boundaries can be efficiently handled using our filtered algorithm Divide and Conquer Technique. The future work is that we mix the different clustering datasets (labeled, unlabeled, nominal, and ordinal) with different algorithms.

References

[1] Ahmad A, Dey L, "A k-mean clustering algorithm for mixed numeric and categorical data", Data and Knowledge Engineering Elsevier Publication, vol. 63, **(2007)**, pp 503-527.

[2] Wang X, Wang X, Wilkes DM, "A Divide-and-Conquer Approach for Minimum Spanning Tree-Based Clustering", IEEE Knowledge and Data Engineering Transactions, vol. 21, **(2009)** July.

[3] Deng S, He Z, Xu X, "Clustering mixed numeric and categorical data: A cluster ensemble approach", Arxiv preprint cs/0509011, **(2005)**.

[4] Guha S, Rastogi R, Shim K, "ROCK: A Robust Clustering Algorithm for Categorical Attributes", Information Systems, vol. 25, no. 5, **(2000)**, pp. 345-366.

[5] Chatzis SP, "A fuzzy c-means-type algorithm for clustering of data with mixed numeric and categorical attributes employing a probabilistic dissimilarity functional", Department of Electrical and Electronic Engineering, Imperial College London, Exhibition Road, South Kensington Campus SW7 2BT, UK, **(2000)**.

[6] Gan G, Yang Z, Wu J, "A Genetic k-Modes Algorithm for Clustering for Categorical Data", ADMA, LNAI 3584, **(2005)**, pp. 195–202.

[7] Han J, Kamber M, "Data Ware Housing and Data Mining. Concepts and Techniques", Third Edition, **(2007)**.

[8] Huang Z, Ng MK, "A fuzzy k-modes algorithm for clustering categorical data", Manage Inf. Principles Ltd., Melbourne, Vic., vol. 7, **(2005)**, pp. 446-452.

[9] Xiong T, Wang S, Mayers A, Monga E, "A New MCA-Based Divisive Hierarchical Algorithm for Clustering Categorical Data", Dept. Comput. Sci., Univ. of Sherbrooke, Sherbrooke, QC, Canada, **(2000)**.

[10] Iam-On N, Boongeon T, Garrett S, Price C, "A Link-Based Cluster Ensemble Approach for Categorical Data Clustering", Aberystwyth University, Aberystwyth, vol. PP 1, **(2000)**.

[11] Izakian H, Abraham A, Snasel V, "Clustering categorical data using a swarm-based method", Machine Intell. Res. Labs. (MIR Labs.), Auburn, WA, USA, **(2000)**, pp. 1720-1724

[12] Aggarwal CC, "Towards Systematic Design of Distance Functions for Data Mining Applications", SIGKDD '03, **(2003)** August 24-27, Washington, DC, USA.

[13] Zhang H, Cao Z, Qiang F, "Representation and clustering of numeric data in concept formation", Dept. of Comput. Sci., China Univ. of Geosci., Wuhan, vol.1, **(2000)**, pp. 597-600.

[14] Mahdavi M, Abolhassani H, "Harmony K-means algorithm for document clustering", Data Min Knowl Disc 2009, vol. 18, **(2009)**, pp. 370–391.

Authors

M. V. Jagannath Reddy received his B E in Electronics and Communication Engg. and M.Tech. in Computer Science Engg. He is pursuing Ph.D in Computer Science and Engineering. He has got 15 years of teaching and industrial experience. He served as the Head, Dept of CSE & IT at MITS, Madanapalle, during the year 2009-2010. His areas of interests include Data Mining and Data warehousing, Intelligent Systems, DBMS. He is a member of ISTE, IAENG and IACSIT. He has published 7 papers in International journals and conferences. Some of his publications appear in IEEE and IJCSIT digital libraries.

Dr. B. Kavitha received the MCA from Sri Venkateswara University, Tirupati and Ph.D Computer Science from Padmavathi Mahila University, Tirupati. She has more than 12 years of teaching experience. Presently she is working has Director, department of MCA, Sree Vidyanikethan Engineering College. Her areas of interest are Fuzzy logic in data bases, Software Engineering, Data warehousing and mining. She published more than 15 papers in international journals and conferences. Some of her publications appear in IEEE and IJCSIT digital libraries.