Volume 5, No. 4, April 2014 (Special Issue)

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

Finding the Optimal Clusters from Discrete Data using Evolutionary Clustering Algorithm

P.M.Chaudhari
Post Graduate Department of Computer Science & Engineering, G. H. Raisoni College of Engineering, Nagpur, India

R.V. Dharaskar
Matoshri Pratishthan's Group of Institutions (MPGI) Integrated Campus, Nanded, India

V. M. Thakare
Post Graduate Department of Computer Science, Faculty of Engineering &Technology, S.G.B. Amravati University, Amravati.India

Abstract: To find the natural number of clusters in a dataset has been a difficult issue in many clustering algorithms. It is complex to determine the desired number of clusters. Even if a hierarchical tree of clusters is given, it is still hard to determine the cut points. The proposed algorithm is modified to find the optimal number of clusters. It is varied in several ways. It allows a variable number of clusters in the chromosomes. It introduces split and merge mutation and new crossover operations. In addition, the fitness function is enhanced to give a fair evaluation. The experimental results illustrate the proposed algorithm can successfully find the correct number of clusters in many datasets.

Keywords: Clustering algorithm, Fitness Function

I. INTRODUCTION

Clustering is also known as unsupervised learning, numerical taxonomy, vector quantization, and learning by observation [1]. Over the last several decades, many clustering algorithms have been developed and used with much success in many applications such as scientific data exploration, information retrieval, text mining, spatial database applications, computer vision, web analysis, marketing, medical diagnostics, and computational biology. More recently, clustering has been used in data mining to deal with data that typically finds in database or data warehouse of large enterprises [2]. These databases or data warehouse typically consist of not just continuous-valued numerical data, but also large amount of data characterized by discrete variables. Given a set of records each characterized by a common set of attributes, clustering is concerned with the grouping together of similar records based on their attribute values.

Given that such data are so commonly found in different application areas such as those in business, medicine and the social sciences, it is important that an effective algorithm can be developed for these data. A discrete variable is a variable that can take on two or more values, for example, colour can take on {red, yellow, green,…} and weather conditions can take on {sunny, rainy cloudy,…}, etc. Many clustering algorithms have been proposed in the past to handle numeric data and relatively little attention have been given to discrete data. Since the proximity measure used in traditional clustering algorithms cannot be defined directly on discrete data, these algorithms do not handle discrete data clustering effectively.

A. Clustering Problem:

When a clustering algorithm has to perform effectively with discrete data, it has to deal with a number of problems including: (i) determining the number of clusters to be formed when given a data set; (ii) handling noisy and missing data values; and (iii) allowing clustering results to be interpreted and (iv) processing capability to deal with discrete data.

a. To determine the number of clusters to be formed:

One problem that faced by many clustering algorithms is to determine how many clusters to form in a given data set. Many popular clustering techniques such as the K-means algorithm, requires that the desired number of clusters to be formed to be decided by the users. Providing such information is often difficult as there is usually a lack of enough domain knowledge for such decision. Because of this difficulty, some effort has been made to look into the determination of the "right" number of clusters [3].

b. To handle difficulties in missing and noisy values:

Real-world data are often incomplete and inconsistent. Missing and noisy values are common in all kinds of databases [4]. Missing values can be found for numerous reasons. For example, it is possible that relevant information may not be available at the time of collection, or may not be regarded as important at the time of entry. They may also be due to human error such as omission. Other than missing values, dirty or noisy data can be recorded as a result of inconsistency of data sources, or randomness or variations when measurement data are obtained. They may also be introduced as a result of deliberately supply of false information. Many clustering techniques do not perform well with the presence of too many noisy values in the data. For

CONFERENCE PAPER
Two day National Conference on Innovation and Advancement in Computing
Organized by: Department of IT, GITAM UNIVERSITY Hyderabad (A.P.) India
Schedule: 28-29 March 2014

example, for centroid based clustering algorithm such as the K-means algorithm, if the initial centers are not chosen properly, the noisy values could cause splitting of a cluster or merging of two supposedly distinct clusters into one. This problem is complicated by the fact that many clustering algorithms do not automatically determine the number of clusters that should be formed. As a result, noisy and missing values can significantly distort the results of some clustering algorithms.

To avoid the problems caused by noisy values, a data set has to be preprocessed by data cleaning or data cleansing procedures [5]. However, no data cleaning method can filter data perfectly and they may not be suitable for all problem domains. For instance, given a clinical database, the missing of some symptoms may actually provide useful information. In this case, the use of typical data cleaning and preprocessing techniques, such as inserting normalized data into the missing field, modifying prior probabilities, or using average data values [6], etc., may grossly distort the original data [7]. For a clustering algorithm to effectively perform its tasks there is a need for it to be able to handle noisy and missing values as much as possible.

c. To interpret the difficulties in clustering results:

Data interpretation is very important in many data mining tasks [8]. However, for clustering, the basis on which data assignments are made is not always clear. It is also not always clear what knowledge different clustering techniques discover. For example, clustering techniques based on the use of distance metrics do not interpret clustering results but merely provide grouping information. Some clustering techniques, such as hierarchical agglomerative clustering techniques [9] do generate hierarchies but does not necessarily allow easy interpretation of the nature of patterns revealed inside the hierarchy. One way to deal with this interpretation problem is to use a separate inductive learning algorithm, such as C4.5 [10] to generate a decision tree that can, hopefully, allow patterns underlying each cluster to be explicitly described. While this can help understand differences between clusters, these inductive learning techniques are normally designed quite differently from the algorithm that generates the clusters.

d. To address difficulties with discrete data:

Many clustering algorithms have mainly been developed to deal with continuous- valued data. Typically, a distance measure defined in the Euclidean space are used to determine if two data records are similar or close enough to be placed in the same clusters. However, distance measure defined in the Euclidean space cannot be used with discrete data that are defined on the nominal scale. For this to be feasible, the discrete attributes have to be binarized, i.e., a binary attribute needs to be created for each unique value of the original discrete attribute. Even with all categories binarized, experiments have shown that many clustering algorithms developed originally for continues-valued data do not handle such data too well.

B. Overview of the proposed algorithm:

To discover clusters in discrete data, we propose a novel algorithm that is based on a simple genetic algorithm (GA).

GAs, which have been developed to solve multi-objective optimization problems, have recently been widely used in different areas and have been shown to be very successful [11]. Here, we make use of GA's powerful ability to perform probabilistic search to try to find optimal cluster groupings. Like other GA based algorithms, our clustering algorithm is as well based on GA operators; we need to decide on an appropriate encoding scheme and a suitable fitness function. For the encoding scheme, we choose to encode a cluster arrangement in each chromosome with each gene encoding a cluster. As for the fitness function, given a set of discrete data and the cluster arrangement as encoded in a chromosome, the fitness function we choose is probabilistic and is based on an information-theoretic entropy measure. It measures the frequency of occurrence of same attribute-value pair within each cluster so that a higher fitness values means the records within the same cluster share more common attribute values. With this fitness function, we aim to optimize the "purity" of the attributes within a cluster.

The fitness measure that is used can be expressed as interesting patterns so that a set of attribute-value pairs can be correlated with a cluster label [12]. For this, one may be attempted to use support and confidence measure that are used to determine if the pattern is interesting as a fitness measure. However, this requires users to define these interestingness measures and this is usually difficult to determine. To overcome this problem, the probabilistic measure the proposed algorithm employs does not require subjective thresholds to be provided. The interestingness of the pattern can be reflected by the value of the function we propose so that the fitter a chromosome is, the more interesting and meaningful the grouping of the chromosome encodes. And this is not affected by the users' choice of interestingness threshold.

In order to maximize the occurrence of an attribute value, it is necessary to regroup data records. This is achieved by special crossover and mutation operators that aim at swapping randomly selecting records from each cluster from one cluster to another. This process is repeated until there is no further improvement in the fitness of the best chromosomes.

In brief, the proposed GA based clustering algorithm has several useful features: (i) it uses an entropy based, rather than the distance based, similarity measure for clustering; (ii) since the fitness measure is probabilistic, it can be used even with noisy and missing values; (iii) it uses a reclassification technique to speed up the identification of optimal solutions; and (iv) it is able to express patterns discovered in each cluster explicitly to allow for them to be better interpreted. In addition to these features, it should be noted that the proposed algorithm could be modified to find a suitable number of clusters that should be discovered in a data set.

II. EVOLUTIONARY CLUSTERING ALGORITHM (ECA) FOR CLUSTERING DISCRETE DATA

In this section, we describe an Evolutionary Clustering Algorithm (ECA) for the clustering of discrete data. Specifically, we describe (i) how different clustering arrangements can be encoded in a population of chromosomes, (ii) a fitness function that can allow the interestingness of

CONFERENCE PAPER
Two day National Conference on Innovation and Advancement in Computing
Organized by: Department of IT, GITAM UNIVERSITY Hyderabad (A.P.) India
Schedule: 28-29 March 2014

different clustering arrangements to be compared; (iii) a crossover operator that can allow interesting patterns discovered in different clustering arrangements to be exchanged; (iv) a mutation operator that can allow variations to clustering arrangements so as to avoid local minimum and (v) a reclassification operator that can correct the wrong clustered records into correct one.

III. EXPERIMENTAL RESULTS

Experiments are set up to examine the effectiveness of ECA to find the natural number of clusters. The natural number of clusters is the number in the class labels in the datasets. Several artificial and real-life datasets will be used for experiments. The parameters of ECA are set as follow. The population is 100.

The maximum iteration is 10000. The crossover rate is 0.06. The swap, split and merge mutation rate will be 0.02, 0.02 and 0.01 respectively. The minimum and maximum numbers of clusters are set to be 2 and 10 accordingly. We examine the ECA with the following datasets 1) Artificial; 2) Soybean ; 3) Vote and 4) Zoo . We run 1000 iterations for each datasets in the experiment. The experimental result is illustrated in Table 1.

Table 1

Name of Dataset	No. of clusters in Dataset	ECA	
		No. of clusters	Accuracy in %
Artificial	4	4	100
Soybean	4	4	100
Vote	2	2	87.13
Zoo	7	4	83.16

Most of the time, it can find the cluster number correctly with high classification accuracy. ECA controls the generation of excessive clusters by two mechanisms. We can increase the merge mutation rate to allow greater possibilities for the cluster to merge together. Also, the reclassification can group back the records according to the interesting patterns obtained. As a result, we can reduce the number of clusters formed. From the experimental results, ECA demonstrates capability on finding the number of clusters.

IV. CONCLUSION

Conventional genetic algorithm has difficulties in solving clustering problem. The proposed algorithm involves a two-phase procedure. In the first phase, it is a genetic-based searching for the meaningful grouping of the data. In the second phase, it applies those interesting patterns to reclassify the records in the dataset. It is altered from records to cluster as the building block for encoding, crossover and mutation.

V. REFERENCES

[1] A. A. Freitas, A Review of Evolutionary Algorithms for Data Mining, In: Soft Computing for Knowledge Discovery and Data Mining, pp. 61-93, O. Maimon; L. Rokach (Editors), Springer, 2007.

[2] J. Handl, J. Knowles, “An Evolutionary Approach to Multi objective Clustering”, IEEE Trans. on Evolutionary Computation, Vol. 11, pp. 56-76, 2007.

[3] D. Jiang, C. Tang, A. Zhang, “Cluster Analysis for Gene Expression Data: A Survey”, IEEE Trans. on Knowledge and Data Engineering, Vol. 16, pp. 1370-1386, 2004.

[4] L. Y. Tseng, S. B. Yang, “A Genetic Approach to the Automatic Clustering Problem”, Pattern Recognition, Vol. 34, pp. 415-424, 2001.

[5] P.M. Chaudhari, R.V. Dharaskar , V.M. Thakare, “Improvement In Post Pareto Analysis In Multi-objective Optimization Using Clustering Technique”, IJPRET, Vol.8, pp 92-99 , April 2013

[6] P.M. Chaudhari, R.V. Dharaskar , V.M. Thakare, “Pareto Analysis in Multi-objective Optimization: An Overview”,IRJCBS, Vol. 9 , December 2012

[7] P.M. Chaudhari, R.V. Dharaskar , V.M. Thakare, “Applying Evolutionary Clustering Technique for finding the most Significant Solution from the Large Result Set obtained in Multi-Objective Evolutionary Algorithms”, IJAMTES, Vol. 7, April 2012

[8] P.M. Chaudhari, R.V. Dharaskar , V.M. Thakare, “Computing the most significant solution from pareto front obtained in multi-objective algorithms” ,IJACSA, Vol. 4, October 2010

[9] P.M. Chaudhari, R.V. Dharaskar , V.M. Thakare, “Application of Clustering Technique for finding the Most Significant Solution from Pareto Front in Multi-objective Evolutionary Algorithms”, CSIT, pp. 93-98, December, 2012

[10] P.M. Chaudhari, R.V. Dharaskar , V.M. Thakare, “Applying Clustering Technique in organizing, classifying and finding the most Significant Solution from the Large Result Set obtained in Multi-Objective Evolutionary Algorithms”, ICEEE, pp. 104-109, October, 2011

[11] P.M. Chaudhari, “Relations of GA operators & Resemblance in Different Crossover Operators - an overview” , XV International Symposium on Theoretical Electrical Engineering ISTET ’09, pp. 383-386, 22-24 June 2009 , Lubeck , Germany

[12] P.M. Chaudhari, “Application of Genetic Algorithms in Structural Representation of Proteins” , First International Conference on Emerging Trends in Engineering and Technology ,pp. 86 , 16-18 July 2008, http://doi.ieeecomputersociety.org/10.1109/ICETET.2008.212

[13] M. Steinbach, L. Ertoz, V.Kumar, , The challenges of clustering high dimensional data, Technical Report , 2001

[14] Y. Zhang, , A. Fu, , C. Cai, , P. Heng. , Clustering discrete data. Proc. of the 16th ICDE, 305, San Diego, CA , 2000

[15] V. S. Alves, R. J. G. B. Campello, E. R. Hruschka, “A Fuzzy Variant of an Evolutionary Algorithm for Clustering”, In Proc. IEEE Int. Conference on Fuzzy Systems, pp. 375-380,

2007.

[16] H. Liu, L. Yu, "Toward Integrating Feature Selection Algorithms for Classification and Clustering", IEEE Trans. on Knowledge and Data Engineering, Vol. 17, pp. 1-12, 2005.

[17] Y. Liu, K. Chen, X. Liao, W. Zhang, "A Genetic Clustering Method for Intrusion Detection", Pattern Recognition, Vol. 37, pp. 927-942, 2004.

[18] Y. Lu, S. Lu, F. Fotouhi, Y. Deng, S. J. Brown, "FGKA: A Fast Genetic K-means Clustering Algorithm", In Proc. ACM Symposium on Applied Computing, pp. 622-623, 2004.

CONFERENCE PAPER
Two day National Conference on Innovation and Advancement in Computing
Organized by: Department of IT, GITAM UNIVERSITY Hyderabad (A.P.) India
Schedule: 28-29 March 2014