International Research Journal in Advanced Engineering and Technology (IRJAET)

ISSN (Print) : 2454-4744 | ISSN (Online) : 2454-4752 (www.irjaet.com)

Vol. 1, Issue 1, pp. 49-53, June, 2015

A STUDY FOR DATA CLUSTERING TECHNIQUES IN DATA MINING APPLICATIONS

S.Aravindhan

Research Scholar, Periyar University, Salem

Dr.D.Maruthanayagam

Assistant Professor, Department of Computer Science, Sri Vijay Vidyalaya College of Arts & Science, Dharmapuri.

ARTICLE INFO

Article History:

Received 1st, Sep, 2015
Received in revised form 8th, Sep, 2015
Accepted 24th, Sep, 2015
Published online 26th, Sep, 2015

***Keywords*: Web mining, clustering, database, data clustering, algorithms and web documents.**

ABSTRACT

Clustering is the subject of active research in several fields such as statistics, pattern recognition, and machine learning. Data mining adds to clustering the complications of very large datasets with very many attributes of different types. This imposes unique computational requirements on relevant clustering algorithms. A variety of algorithms have recently emerged that meet these requirements and were successfully applied to real-life data mining

1. Introduction

The goal of this study is to provide a comprehensive review of different clustering techniques in data mining. Clustering is a division of data into groups of similar objects. Each group, called cluster, consists of objects that are similar between themselves and dissimilar to objects of other groups. Representing data by fewer clusters necessarily loses certain fine details but achieves simplification. It represents many data objects by few clusters, and hence, it models data by its clusters. Data modeling puts clustering in a historical perspective rooted in mathematics, statistics, and numerical analysis. From a machine learning perspective clusters correspond to hidden patterns, the search for clusters is unsupervised learning, and the resulting system represents a data concept. Therefore, clustering is unsupervised learning of a hidden data concept. Data mining deals with large databases that impose on clustering analysis additional severe computational

requirements. These challenges led to the emergence of powerful broadly applicable data mining clustering methods [1].

Data Clustering is considered an interesting approach for finding similarities in data and putting similar data into groups. Clustering partitions a data set into several groups such that the similarity within a group is larger than that among groups [2]. The idea of data grouping, or clustering, is simple in its nature and is close to the human way of thinking; whenever we are presented with a large amount of data, we usually tend to summarize this huge number of data into a small number of groups or categories in order to further facilitate its analysis. Moreover, most of the data collected in many problems seem to have some inherent properties that lend themselves to natural groupings. Nevertheless, finding these groupings or trying to categorize the data is not a simple task for humans unless the data is of low dimensionality.

This is why some methods in soft computing have been proposed to solve this kind of problem. Those methods are called "Data Clustering Methods" and they are the subject of this paper. Clustering algorithms are used extensively not only to organize and categorize data, but are also useful for data compression and model construction. By finding similarities in data, one can represent similar data with fewer symbols for example. Also if we can find groups of data, we can build a model of the problem based on those groupings. Another reason for clustering is to discover relevance knowledge in data. Francisco Azuaje *et al.* [3] implemented a Case Based Reasoning (CBR) system based on a Growing Cell Structure (GCS) model. Data can be stored in a knowledge base that is indexed or categorized by cases; this, is what is called a Case Base. Each group of cases is assigned to a certain category. Using a Growing Cell Structure (GCS) data can be added or removed based on the learning scheme used.

Later when a query is presented to the model, the system retrieves the most relevant cases from the case base depending on how close those cases are to the query.

2. Data Clustering Overview

In this section, four of the most representative off-line clustering techniques are reviewed:

A. K-means (or Hard C-means) Clustering,
B. Fuzzy C-means Clustering,
C. Mountain Clustering, and
D. Subtractive Clustering.

These techniques are usually used in conjunction with radial basis function networks (RBFNs) and Fuzzy Modeling. The first technique is ***K-means clustering*** [4] (or *Hard C-means* clustering, as compared to *Fuzzy C-means* clustering.) This technique has been applied to a variety of areas, including image and speech data compression [5, 6] data preprocessing for system modeling using radial basis function networks, and task decomposition in heterogeneous neural network architectures [7]. This algorithm relies on finding cluster centers by trying to minimize a cost function of dissimilarity (or distance) measure.

The second technique is ***Fuzzy C-means*** clustering, which was proposed by Bezdek in 1973 [1] as an improvement over earlier Hard C-means clustering. In this technique each data point belongs to a cluster to a degree specified by a membership grade. As in K-means clustering, Fuzzy C-means clustering relies on minimizing a cost function of dissimilarity measure.

The third technique is ***Mountain clustering***, proposed by Yager and Filev [1]. This technique
builds calculates a mountain function (density function) at every possible position in the data space, and chooses the position with the greatest density value as the center

of the first cluster. It then destructs the effect of the first cluster mountain function and finds the second cluster center. This process is repeated until the desired numbers of clusters have been found.

The fourth technique is ***Subtractive clustering***, proposed by Chiu [1]. This technique is similar to mountain clustering, except that instead of calculating the density function at every possible position in the data space, it uses the positions of the data points to calculate the density function, thus reducing the number of calculations significantly.

3. Data Clustering Techniques

In this section a detailed discussion of each technique is presented.

A.K-means Clustering

The K-means clustering, or Hard C-means clustering, is an algorithm based on finding data clusters in a data set such that a cost function (or an objection function) of dissimilarity (or distance) measure is minimized [1]. In most cases this dissimilarity measure is chosen as the Euclidean distance.

The algorithm is presented with a data set, *Xi, i = 1…, n;* it then determines the cluster centers Ci and the membership matrix **U** iteratively using the following steps:

Step 1: Initialize the cluster center *Ci, i= 1…, c* .This is typically done by randomly selecting *c* points from among all of the data points.

Step 2: Determine the membership matrix **U** by Equation (2).

Step 3: Compute the cost function according to Equation (1). Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.

Step 4: Update the cluster centers according to Equation (3). Go to step 2.

The performance of the K-means algorithm depends on the initial positions of the luster enters, thus it is advisable to run the algorithm several times, each with a different set of initial cluster centers [8, 9].

B. Fuzzy C-means Clustering

Fuzzy C-means clustering (FCM), relies on the basic idea of Hard C-means clustering (HCM), with the difference that in FCM each data point belongs to a cluster to a degree of membership grade, while in HCM every data point either belongs to a certain cluster or not. So FCM employs fuzzy partitioning such that a given data point can belong to several groups with the degree of belongingness specified by membership grades between 0 and 1. However, FCM still uses a cost function that is to be minimized while trying to partition the data set.

$$\sum_{i=1}^{c} u_{ij} = 1, \quad \forall j = 1, \ldots, n$$

------- (1)

$$J(U, \mathbf{c}_1, \ldots, \mathbf{c}_c) = \sum_{i=1}^{c} J_i = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d_{ij}^2$$

--------- (2)

$$\mathbf{c}_i = \frac{\sum_{j=1}^{n} u_{ij}^m \mathbf{x}_j}{\sum_{j=1}^{n} u_{ij}^m}$$

---------- (3)

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d_{ij}}{d_{kj}} \right)^{2/(m-1)}}$$

---------- (4)

The algorithm works iteratively through the preceding two conditions until the no more improvement is noticed. In a batch mode operation, FCM determines the cluster centers *Ci* and the membership matrix **U** using the following steps:

Step 1: Initialize the membership matrix **U** with random values between 0 and 1 such that the constraints in Equation (1) are satisfied.
Step 2: Calculate *c* fuzzy cluster centers, *Ci, i= 1…, c* , using Equation (3).
Step 3: Compute the cost function according to Equation (2). Stop if either it is below a certain tolerance value or its improvement over previous iteration is below a certain threshold.
Step 4: Compute a new **U** using Equation (4). Go to step 2.

As in K-means clustering, the performance of FCM depends on the initial membership matrix values; thereby it is advisable to run the algorithm for several times, each starting with different values of membership grades of data points [9].

C. Mountain Clustering

The mountain clustering approach is a simple way to find cluster centers based on a density measure called the *mountain function*. This method is a simple way to find approximate cluster centers, and can be used as a preprocessor for other sophisticated clustering methods.
Step 1: It involves forming a grid on the data space, where the intersections of the grid lines constitute the potential cluster centers, denoted as a set V.
Step 2: It entails constructing a mountain function representing a data density measure. The height of the mountain function at a point **v** □ V *is* equal to

$$m(\mathbf{v}) = \sum_{i=1}^{N} \exp\left(-\frac{\|\mathbf{v}-\mathbf{x}_i\|^2}{2\sigma^2} \right)$$

---------- (5)

Step 3: It involves selecting the cluster centers by sequentially destructing the mountain function. The first cluster center C**1** is determined by selecting the point with the greatest density measure. Obtaining the next cluster center requires eliminating the effect of the first cluster. This is done by revising the mountain function: a new mountain function is formed by subtracting a scaled Gaussian function centered at C**1:**

$$m_{\text{new}}(\mathbf{v}) = m(\mathbf{v}) - m(\mathbf{c}_1)\exp\left(-\frac{\|\mathbf{v}-\mathbf{c}_1\|^2}{2\beta^2} \right)$$

----

(6)

The subtracted amount eliminates the effect of the first cluster. Note that after subtraction, the new mountain function new *mnew* (**v**) reduces to zero at v = C**1**
Step 4: After subtraction, the second cluster center is selected as the point having the greatest value for the new mountain function. This process continues until a sufficient number of cluster centers are attained [9].

D. Subtractive Clustering

The problem with the previous clustering method, mountain clustering, is that its computation grows exponentially with the dimension of the problem; that is because the mountain function has to be evaluated at each grid point. Subtractive clustering solves this problem by using data points as the candidates for cluster centers, instead of grid points as in mountain clustering. This means that the computation is now proportional to the problem size instead of the problem dimension. However, the actual cluster centers are not necessarily located at one of the data points, but in most cases it is a good approximation, especially with the reduced computation this approach introduces [9].
Since each data point is a candidate for cluster centers, a *density measure* at data point *i* **x** is defined as

$$D_i = \sum_{j=1}^{n} \exp\left(-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|^2}{(r_a/2)^2} \right)$$

---------- **(7)**

Where *ra* is a positive constant representing a neighborhood radius. Hence, a data point will have a high density value if it has many neighboring data points. The first cluster center X*C1* is chosen as the point having the largest density value D*C1*. Next, the density measure of each data point **Xi** is revised as follows:

$$D_i = D_i - D_{c_1} \exp\left(-\frac{\left\| \mathbf{x}_i - \mathbf{x}_{c_1} \right\|^2}{(r_b / 2)^2} \right)$$

-------- (8)

Where *rb* is a positive constant which defines a neighborhood that has measurable reductions in density measure. Therefore, the data points near the first cluster center X*C1* will have significantly reduced density measure.

After revising the density function, the next cluster center is selected as the point having the greatest density value. This process continues until a sufficient number of clusters are attainted.

4. Conclusion

Four clustering techniques have been reviewed in this paper, namely: K-means clustering, Fuzzy C-means clustering, Mountain clustering, and Subtractive clustering. These approaches solve the problem of categorizing data by partitioning a data set into a number of clusters based on some similarity measure. so that the similarity in each cluster is larger than among clusters. The comparative study done here is concerned with the accuracy of each algorithm, with care being taken toward the efficiency in calculation and other performance measures. Finally, the clustering techniques discussed here and also they can be used in conjunction with other neural or fuzzy systems for further refinement of the overall system performance.

References:

[1] **"**Survey of Clustering Data Mining Techniques", Pavel BerkhinAccrue Software, Inc.

[2] Jang, J.-S. R., Sun, C.-T., Mizutani, E., "Neuro-Fuzzy and Soft Computing – A Computational Approach to Learning and Machine Intelligence, *"Prentice Hall.*

[3] Azuaje, F., Dubitzky, W., Black, N., Adamson, K.,"Discovering Relevance Knowledge in Data: A Growing Cell Structures Approach," *IEEETransactions on Systems, Man, and Cybernetics- Part B: Cybernetics, Vol. 30, No. 3, June 2000 (pp.448).*

[4] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," Applied Statistics, 28:100--108, 1979.

[5] Lin, C., Lee, C., "Neural Fuzzy Systems," *Prentice Hall, NJ, 1996.*

[6] Tsoukalas, L., Uhrig, R., "Fuzzy and Neural Approaches in Engineering," *John Wiley & Sons, Inc., NY, 1997.*

[7] Nauck, D., Kruse, R., Klawonn, F., "Foundations of Neuro-Fuzzy Systems," *John Wiley & Sons Ltd., NY, 1997.*

[8] J. A. Hartigan and M. A. Wong, "A k-means clustering algorithm," Applied Statistics, 28:100--108, 1979.

[9] "A Comparative Study of Data Clustering Techniques", Khaled Hammouda, Prof. Fakhreddine Karray, *University of Waterloo, Ontario, Canada*