Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 11, November 2014, pg.422 – 428

RESEARCH ARTICLE

A Comparative Study of Various Clustering Algorithms in Data Mining

S.Saraswathi[1], Dr. Mary Immaculate Sheela [2]

[1]Research Scholar, Research & Development Centre, Bharathiar University, Coimbatore, India

[2]Professor, Computer Science and Engineering Department, R.M.D. Engineering College, Kavaraipettai, Chennai

[1] sararavi2001@gmail.com, [2] drsheela09@gmail.com

Abstract - Data mining is the process of extracting Knowledge from data. Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group are more similar to each other than to those in other groups. Clustering is one of the complicated tasks in data mining. It plays a vital role in a broad range of applications such as marketing, surveillance, fraud detection, Image processing, Document classification and scientific discovery. Lot of issues related with cluster analysis such as a high dimension of the dataset, arbitrary shapes of clusters, scalability, input parameter, complexity and noisy data are still under research. A variety of algorithms have been emerged for clustering to address these issues which causes perplexity in choosing the right algorithm for research applications. This paper deals with classification of some of the well known clustering algorithms and also their comparison based on key issues, advantages and disadvantages, which provide guidance for the selection of clustering algorithm for a specific application.

Keywords - Clustering algorithms, Partitioning methods, Hierarchical methods and Density based methods

---

I. INTRODUCTION

Data mining is truly an interdisciplinary topic that can be defined in many different ways. In the field of database management industry, data analysis is mainly evolved with number of large data repositories. The result yields to the process of data mining. There are a number of data mining functionalities used to specify the kinds of patterns to be found in data mining task. These functionalities include characterizations and discrimination, the mining of frequent patterns, associations and correlations, classification regression, clustering analysis and outlier analysis[1]. Clustering is one of the most interesting and important topics in data mining that aims to find intrinsic structures in data, and organize them into meaningful subgroups for further study and analysis. The basic concept of cluster analysis is partitioning a set of data objects or observations into subsets. Each subset is unique such that objects in one cluster are similar to one another, yet dissimilar to objects in other cluster.

Different cluster may be formed using same data set by applying different clustering methods[2]. The clustering is more challenging task than classification. High dimension of the dataset, arbitrary shapes of clusters, scalability, input parameter, domain knowledge and handling of noisy data are some of the basic requirement for cluster analysis. There are many well established clustering algorithm are present in literature. This makes a great challenge for the user to do selection among the available algorithm for the specific task. In this paper we discuss some of the popular clustering algorithms and also an attempt has been made to provide guidance for the selection of clustering algorithm for a specific application to the user.

II. CLASSIFICATION OF CLUSTERING ALGORITHMS

With the advent of technology, a lot of clustering algorithms with peculiar features were proposed and it is difficult to categorize them with a solid boundary. Even then clustering algorithms can be broadly classified into three categories according to their working principle as Partitioning methods, Hierarchical methods, Density based methods.

In short, partitioning algorithms attempt to determine k clusters that optimize a certain, often distance-based criterion function. Hierarchical algorithms create a hierarchical decomposition of the database that can be presented as a dendrogram. Density-based algorithms search for dense regions in the data space that are separated from one another by low density noise regions.

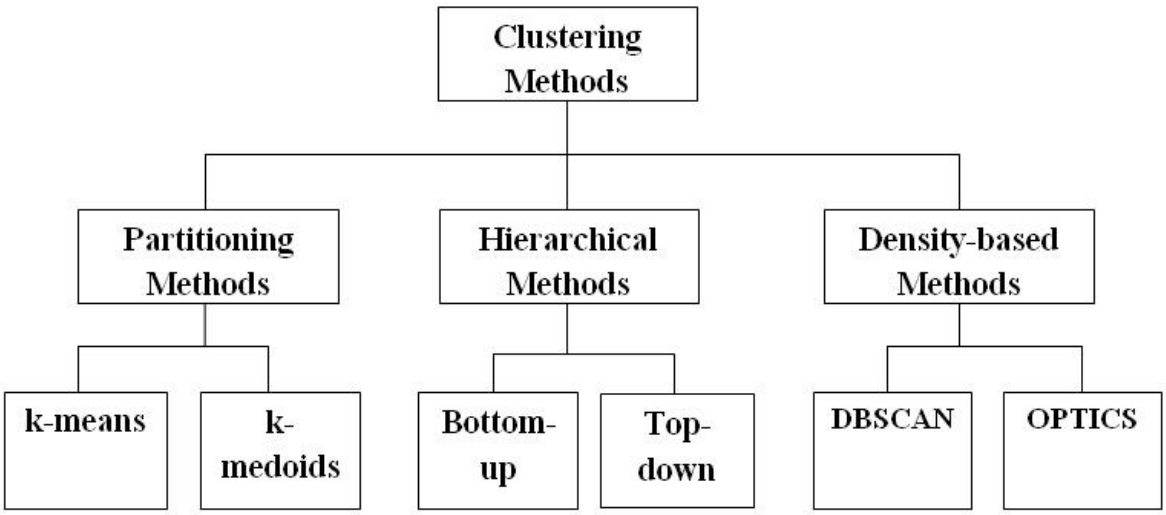

Figure 1 : Clustering Algorithms Classification

A. Partitioning Clustering Algorithms

Partitioning method conducts one - level partitioning on data set, first it creates initial set of k partition, where parameter k is the number of partition to construct. It then uses an iterative relocation technique that attempts to improve the partitioning by moving objects from one group to another group. Typical partitioning method includes two popular algorithms, k - means and k - medoids [4]. Typically, k seeds are randomly selected and then a relocation scheme iteratively reassigns points between clusters to optimize the clustering criterion. [3] The minimization of the square-error criterion - sum of squared Euclidean distances of points from their closest cluster centroid, is the most commonly used. A serious drawback of partitioning algorithms is that there are a number of possible solutions.

1) K-Means : K - means clustering is a partitioning method. K - means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean [5]. Despite its wide popularity, k-means is very sensitive to noise and outliers since a small number of such data can substantially influence the centroids. The weakness are sensitivity to initialization, entrapments into local optima, poor cluster descriptors, inability to deal with clusters of arbitrary shape, size and density, reliance on user to specify the number of clusters.

It proceeds as follows:

1. Randomly selects k of the objects, each of which initially represents a cluster mean or center.
2. For each of the remaining objects, an object is assigned to the cluster to which it is the most similar, based on the distance between the object and the cluster mean.
3. It then computes the new mean for each cluster. This process iterates until the criterion function converges. Typically, the square-error criterion is used, defined as

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} |p - m_i|^2$$

where E is the sum of the square error for all objects in the data set; p is the point in space representing a given object; and m_i is the mean of cluster Ci (both p and m_i are multidimensional). In other words, for each object in each cluster, the distance from the object to its cluster center is squared, and the distances are summed.

The k – mean algorithm means algorithm has the following important properties:

1. It is efficient in processing large data sets.
2. It often terminates at a local optimum.
3. It works only on numeric values.
4. The clusters have convex shapes.

2) K-Medoids : The partitioning algorithm in which cluster is represented by one of the objects located near its centre is called as a k- mediods. PAM, CLARA and CLARANS are three main algorithms proposed under the k-mediod method [6]. The k - means algorithm is sensitive to outliers because an object with an extremely large value may substantially distort the distribution of data. Instead of taking the mean value of the objects in a cluster as a reference point, we can pick actual objects to represent the clusters, using one representative object per cluster. Each remaining object is clustered with the representative object to which it is the most similar. The partitioning method is then performed based on the principle of minimizing the sum of the dissimilarities between each object and its corresponding reference point [4]. [1] An absolute-error criterion is used, defined as

$$E = \sum_{j=1}^{k} \sum_{p \in C_j} |p - o_j|$$

where E is the sum of the absolute error for all objects in the data set; p is the point in space representing a given object in cluster C_j ; and o_j is the representative object of C_j . In general, the algorithm iterates until, eventually, each representative object is actually the medoid, or most centrally located object, of its cluster. This is the basis of the k-medoids method for grouping n objects into K clusters. The iterative process of replacing representative objects by non representative objects continues as long as the quality of the resulting clustering is improved.

B. Hierarchical Algorithms

As the name implies, the hierarchical methods, tries to decompose the dataset of n objects into a hierarchy of a groups. This hierarchical decomposition can be represented by a tree structure diagram called as a *dendrogram*; whose root node represents the whole dataset and each leaf node is a single object of the dataset. The clustering results can be obtained by cutting the dendrogram at different level. There are two general approaches for the hierarchical method: agglomerative (bottom-up) and divisive (top down) [6].

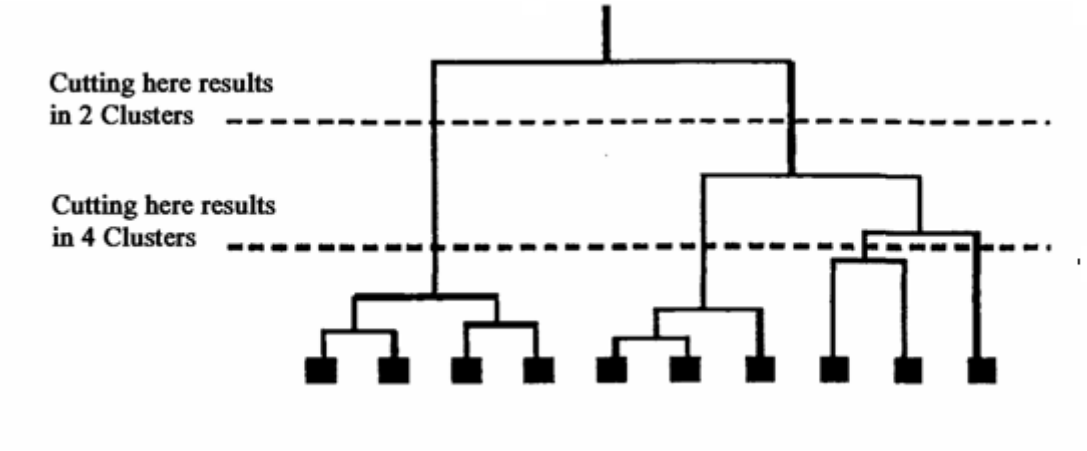

Figure 2: Hierarchical Clustering

The merging or splitting stops once the desired number of clusters has been formed. Typically, each iteration involves merging or splitting a pair of clusters based on a certain criterion, often measuring the proximity between clusters. Hierarchical techniques suffer from the fact that previously taken steps (merge or split), possibly erroneous, are irreversible [3]. The Representative algorithms proposed for hierarchical concept are CURE, BIRCH and CHAMELEON.

The construction of a hierarchical agglomerative classification can be achieved by the following general algorithm.
1. Find the 2 closest objects and merge them into a cluster
2. Find and merge the next two closest points, where a point is either an individual object or a cluster of objects.
3. If more than one cluster remains ,return to step 2
.

1) CURE : Clustering Using Representatives (CURE) is an agglomerative method introducing two novelties. First, clusters are represented by a fixed number of well-scattered points instead of a single centroid. Second, the representatives are shrunk toward their cluster centers by a constant factor. At each iteration, the pair of clusters with the closest representatives is merged [3]. CURE is capable of finding clusters of different shapes and sizes, and it is insensitive to outliers. Because CURE uses sampling, estimation of its complexity is not straightforward. It also uses two techniques to achieve scalability: data sampling, and data partitioning [7].

2) BIRCH : One of the most striking developments in hierarchical clustering is the algorithm BIRCH. **BIRCH** (balanced iterative reducing and clustering using hierarchies) is an unsupervised data mining algorithm used to perform hierarchical clustering over particularly large data-sets. An advantage of BIRCH is its ability to incrementally and dynamically cluster incoming, multi-dimensional metric data points in an attempt to produce the best quality clustering for a given set of resources. [3] It introduces a novel hierarchical data structure, CF-tree, for compressing the data into many small sub-clusters and then performs clustering with these summaries rather than the raw data. Sub-clusters are represented by compact summaries, called cluster-features (CF) that are stored in the leafs. The non-leaf nodes store the sums of the CF of their children. A CF-tree is built dynamically and incrementally, requiring a single scan of the dataset. An object is inserted in the closest leaf entry. [1]Two input parameters control the maximum number of children per non-leaf node and the maximum diameter of sub-clusters stored in the leafs. Once the CF-tree is built, any partitioning or hierarchical algorithms can use it to perform clustering in main memory. BIRCH is reasonably fast, but has two serious drawbacks: data order Sensitivity and inability to deal with non-spherical clusters of varying size because it uses the concept of diameter to control the boundary of a cluster.

3) CHAMELEON : Chameleon is a hierarchical clustering algorithm that uses dynamic modeling to determine the similarity between pairs of clusters. In Chameleon, cluster similarity is assessed based on how well-connected objects are within a cluster and on the proximity of clusters. That is, two clusters are merged if their interconnectivity is high and they are close together. Chameleon has been shown to have greater power at discovering arbitrarily shaped clusters of high quality than

several well-known algorithms such as BIRCH and density based DBSCAN.[3] Due to its dynamic merging model CHAMELEON is more effective than CURE in discovering arbitrary-shaped clusters of varying density. However, the improved effectiveness comes at the expense of computational cost that is quadratic in the database size. However, the processing cost for high-dimensional data may require O(n2) time for n objects in the worst case.

C. Density-Based Methods

To discover clusters with arbitrary shape, density-based clustering methods have been developed. These typically regard clusters as dense regions of objects in the data space that are separated by regions of low density representing noise. An open set in the Euclidean space can be divided into a set of its connected components. The implementation of this idea for partitioning of a finite set of points requires concepts of density, connectivity and boundary. They are closely related to a point's nearest neighbours. [7] A cluster, defined as a connected dense component, grows in any direction that density leads. Therefore, density-based algorithms are capable of discovering clusters of arbitrary shapes. Also this provides a natural protection against outliers. They also have good scalability. These outstanding properties are tempered with certain inconveniencies. From a very general data description point of view, a single dense cluster consisting of two adjacent areas with significantly different densities (both higher than a threshold) is not very informative. Another drawback is a lack of interpretability. [1] There are two major approaches for density-based methods. The first approach pins density to a training data point and the representative algorithms include DBSCAN and OPTICS. The second approach pins density to a point in the attribute space and It includes the algorithm DENCLUE.

1) DBSCAN : DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density based clustering algorithm. The algorithm grows regions with sufficiently high density into clusters and discovers clusters of arbitrary shape in spatial databases with noise. It defines a cluster as a maximal set of density-connected points. [3] A density-based cluster is a set of density-connected objects that is maximal with respect to density-reachability. Every object not contained in any cluster is considered to be noise. This method is sensitive to its parameter e and MinPts, and leaves the user with the responsibility of selecting parameter values that will lead to the discovery of acceptable clusters. If a spatial index is used, the computational complexity of DBSCAN is O(nlogn), where n is the number of database objects. Otherwise, it is O(n2).

2) OPTICS : OPTICS(Ordering Points To Identify the Clustering Structure) computes [1] an augmented cluster ordering for automatic and interactive cluster analysis. Because of the structural equivalence of the OPTICS algorithm to DBSCAN, the OPTICS algorithm has the same runtime complexity as that of DBSCAN, that is, O(nlogn) if a spatial index is used, where n is the number of objects.

3) DENCLUE : DENCLUE (DENsity-based CLUstEring) is a clustering method based on a set of density distribution functions. [1] The method is built on the following ideas: (1) the influence of each data point can be formally modeled using a mathematical function, called an influence function, which describes the impact of a data point within its neighbourhood; (2) the overall density of the data space can be modeled analytically as the sum of the influence function applied to all data points; and (3) clusters can then be determined mathematically by identifying density attractors, where density attractors are local maxima of the overall density function.

III. Comparative Study

Clustering is a challenging task in data mining. There are large number of clustering algorithms, each to solve some specific problem. No clustering algorithm can adequately handle all sorts of cluster structure and input data. The goal of this comparative study is to provide a comprehensive review of different clustering techniques in data mining. It is given in table I. An exhaustive comparative study of different clustering algorithms proposed under the different methods by considering the different aspects of clustering is given in table II. In table II we had provided the comments for each of the algorithm which gives the clear idea of the pros and cons of each of the algorithms.

Clustering Method	Advantages	Disadvantages
Partitioning	• Relatively scalable and simple. • Suitable for datasets with compact spherical clusters that are well-separated.	• Degradation in high dimensional spaces. • Poor cluster descriptors • High sensitivity to initialization phase, noise and outliers
Hierarchical	• Embedded flexibility regarding the level of granularity. • Well suited for problems involving point linkages, e.g. taxonomy trees. • Application to any attribute types	• Inability to make corrections once the splitting/merging decision is made. • Lack of interpretability regarding the cluster descriptors. • Vagueness of termination criterion. • Prohibitively expensive for high dimensional and massive datasets.
Density based	• Discovery of arbitrary-shaped clusters with varying size • Resistance to noise and outliers	• High sensitivity to the setting of input parameters • Poor cluster descriptors • Unsuitable for high-dimensional datasets

Table I : Advantages & Disadvantages of Clustering Methods

Clust. Algorithm	Data Type Handling	Cluster Shape	Complexity	Handling High Dimensional Data	Comments
K-Means	Numeric	Convex	O(nkt)	No	1. Ease of implementation, Simplicity & Efficiency 2. Sensitive to noise 3. Can 't handle clusters of different size
CURE	Numeric	Arbitrary	$O(N^2_{sample}logN_{sample})$	Yes	1. Uses multiple Representatives 2. Improves the Scalability
BIRCH	Numeric	Convex	O(n)	No	1. Designed for clustering a large amount of numerical data. 2. works well only for spherical clusters
CHEMELEON	Numeric	Arbitrary	$O(n^2)$	No	1. More effective than CURE. 2. Produce high quality clusters.

DBSCAN	Discrete	Arbitrary	O(nlogn)	No	1. Can handle noise 2. Can 't handle clusters of different size
OPTICS	Numeric	Arbitrary	O(nlogn)	Yes	1. No need for input parameter settings 2. Can not handle clusters of different densities
DENCLUE	Numeric	Arbitrary	$O(n^2)$	Yes	1. Clusters can be determined mathematically by identifying density attractors. 2. Has good clustering properties for data sets with large amounts of noise.

Table II: Comparative Study of several clustering algorithms

IV. Conclusion

Cluster Analysis is a process of grouping the objects, called as a cluster/s, which consists of the objects that are similar to each other in a given cluster and dissimilar to the objects in other cluster. Cluster analysis, primitive exploration with little or no prior knowledge, consists of research developed across a wide variety of communities. The diversity, on one hand, equips us with many tools. On the other hand, the profusion of options causes confusion. Large number of clustering algorithms had been proposed which satisfy certain key issues such as arbitrary shapes, high dimensional database and domain knowledge and so on. It is not possible to design a single clustering algorithm which fulfils all the requirements of clustering. So it is very difficult to select any algorithm for a specific application. In this paper we give detail about classification of clustering techniques with the advantages and disadvantages. We also tried to provide a detailed comparison of the clustering algorithms and we provided comments on each algorithm which makes the selection process easier for the user.

REFERENCES

[1] Jiawei Han and Michheline Kamber,Data mining concepts and techniques-a reffrence book

[2] X. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, P.S. Yu, Z.H. Zhou, M. Steinbach, D.J.Hand, and D. Steinberg, ―Top 10 Algorithms in Data Mining, ―Knowledge Information Systems, vol. 14, no. 1, pp. 1 37, 2007

[3] Deepti Sisodia, Lokesh Singh, Sheetal Sisodia, Khushboo saxena, Clustering Techniques: A Brief Survey of Different Clustering Algorithms, International Journal of Latest Trends in Engineering and Technology (IJLTET). Vol. 1 Issue 3 September 2012 .

[4] Yaminee S. Patil, M.B.Vaidya , A Technical Survey on Cluster Analysis in Data Mining, International Journal of Emerging Technology and Advanced Engineering Website: www.ijetae.com (ISSN 2250 - 2459, Volume 2, Issue 9, September 2012)

[5] M.Vijayalakshmi, M.Renuka Devi, A Survey of Different Issue of Different clustering Algorithms Used in Large Datasets, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 3, March 2012.

[6] Prof. Neha Soni1, Dr. Amit Ganatra, Comparative study of several Clustering Algorithms, *International Journal of Advanced Computer Research, Volume-2 Number-4 Issue-6 December-2012*

[7] Pavel Berkhin, A Survey of Clustering Data Mining Techniques, Yahoo!, Inc. pberkhin@yahoo-inc.com

