AN OPTIMIZED SURVEY OF CLUSTERING ALGORITHMS

Swapnil Ahire[1], Lakshita Landge[2]

[1] *Student, Computer Science and Engineering, AITR, Madhya Pradesh, India*
[2] *Assistant Professor, Computer Science and Engineering, AITR, Madhya Pradesh, India*

ABSTRACT

The goal of this survey is to provide a comprehensive review and comparison of different clustering techniques in data mining. Clustering is a significant task in data analysis and data mining applications. It is the task of arranging a set of objects so that objects in the identical group are more related to each other than to those in other groups (clusters). Mining can be done by using supervised learning and unsupervised learning. The clustering is unsupervised learning. A good clustering method will produce high superiority clusters with high intra-class similarity and low inter-class similarity. Clustering algorithms can be categorized into partition-based algorithms like K-Means clustering, hierarchical-based algorithms and density-based algorithms like DBSCAN. Partitioning clustering algorithm splits the data points into k partition, where each partition represents a cluster. Partitioning is the centroid based clustering; the value of k-mean is specified priori. Hierarchical clustering is a technique of clustering. It divides the same dataset by constructing a hierarchy of clusters. Density based algorithm (DBSCAN) find the cluster according to the regions which grow with high density. In this survey paper, an analysis of clustering and its different techniques in data mining are studied.

Keyword: - *data mining, clustering, clustering algorithm.*

1. INTRODUCTION

Clustering is an important technique in data mining and it is the process of partitioning data into a set of clusters such that each object in a cluster is similar to another object in the same cluster and dissimilar to every object not in the same cluster. Dissimilarities and similarities are assessed based on the attribute values describing the objects and often involve distance measures. Clustering analyses the data objects without consulting a known class label. This is because the class labels are not known in the first place, and clustering is used to find those labels. Good clustering exhibits high intra-class similarity and low inter-class similarity, that is, the higher the similarity of objects in a given cluster, gives better clustering. The superiority of a clustering algorithm depends equally on the similarity measure used by the method and its implementation. The superiority also depends on the algorithm's ability to find out some or all of the hidden patterns. The different ways in which clustering methods can be compared are partitioning criteria, separation of clusters, similarity measures and clustering space [1].

The traditional way to treat categorical attributes as numeric does not always produce meaningful results because many categorical domains are not ordered. This paper gives a survey of clustering like hierarchical clustering, partition based clustering and algorithms DBSCAN, K-means clustering, BIRCH.

2. DENSITY BASED CLUSTERING

2.1 DBSCAN Algorithm

Algorithm DBSCAN (Density Based Spatial Clustering of Applications with Noise) based on density-based a cluster which is designed to discover clusters of arbitrary shape. [2]

DBSCAN is a density-based algorithm which uses density as number of points in a mentioned radius where point is a core point. This density-based algorithm eliminates noise points and makes each group of connected core points into a separate cluster.

The DBSCAN algorithm was first introduced by Ester, et al. [Ester1996], and relies on a density-based notion of clusters. Clusters are spotted by looking at the density of points. Regions which are having high density of points depict the existence of clusters and the regions with a low density of points indicate clusters of outliers or clusters of noise. This algorithm is suited to deal with large datasets with noise and is able to find clusters with different sizes and shapes. The key idea of the DBSCAN algorithm is, for each point of a cluster, the neighbourhood of a specified radius has to contain at least a minimum number of points and that is, the density in the neighbourhood has to cross some predefined threshold. This algorithm needs three input parameters:

- k, the neighbour list size;
- MinPts, the minimum number of points that should exist in the Eps-neighbourhood;
- Eps, the radius that delimitate the neighbourhood area of a point (Eps neighbourhood).

The clustering process is based on the classification of the points in the dataset as core points, noise points and border points, and also based on the use of density relations between all points(directly density-reachable, density-reachable, density-connected [Ester1996]) to form the clusters. [3]

2.1.1 Algorithmic steps for DBSCAN clustering

1) Start with an arbitrary starting point that has not been visited.
2) Extract the neighbourhood of this point using ε (All points which are within the ε distance are neighbourhood).
3) If there are sufficient neighbourhood around this point then the clustering process starts and point is marked as visited else this point is called as noise. (This point will become the part of the cluster).
4) If a point is found as part of the cluster then its ε neighbourhood is also the part of the cluster and the above procedure from step 2 is repeated for all ε neighbourhood points. Repeat this until all points in the cluster is determined.
5) A new unvisited point is retrieved and processed; it leads to the discovery of a further cluster or noise.
6) This process to be continued until all points are marked as visited.

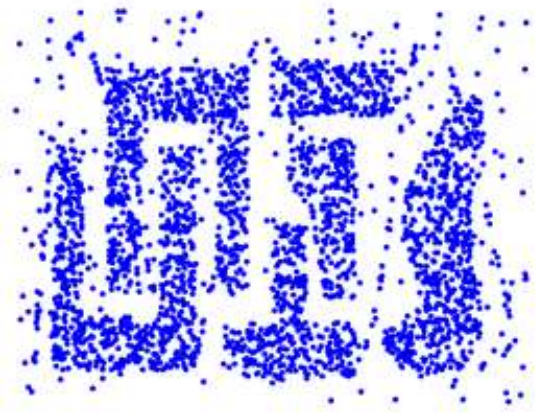

Fig -1: Original Points

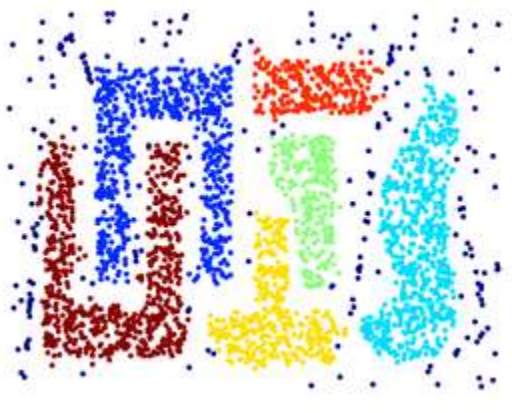

Fig -2: After clustering with DBSCAN.

2.2 Advantages of DBSCAN algorithm

1) Resistant to Noise.
2) Can handle clusters of different shapes and sizes.

2.3 Disadvantages of DBSCAN algorithm

1) DBSCAN has problem of identifying clusters of varying densities.
2) Algorithm can't work well with high dimensional dataset.

3. HIERARCHICAL CLUSTERING ALGORITHM

Hierarchical clustering is categorized into agglomerative (bottom-up) and divisive (top-down). An agglomerative clustering starts with singleton clusters and recursively merges two or more most appropriate clusters.

The Divisive clustering starts with one cluster of all data points and then recursively splits the most appropriate cluster. This process to be continued until a stopping criterion (frequently, the requested number k of clusters) is achieved. [4]

3.1 Agglomerative (AGNES) algorithm

This algorithm maintains an "active set" of clusters and at each stage decides which two clusters to merge. When two clusters are merged, they are each removed from the active set and their union is added to the active set. This iterates until there is only one cluster in the active set. The tree is formed by keeping track of which clusters were merged. [5]

The algorithm forms clusters in bottom-up fashion, as follows:

1. Initially, put each article in its own cluster.
2. Among all current clusters, pick the two clusters with the smallest distance.
3. Replace these two clusters with a new cluster is formed by merging the two original ones.
4. Repeat the above two steps until there is only one remaining cluster in the pool. [6]

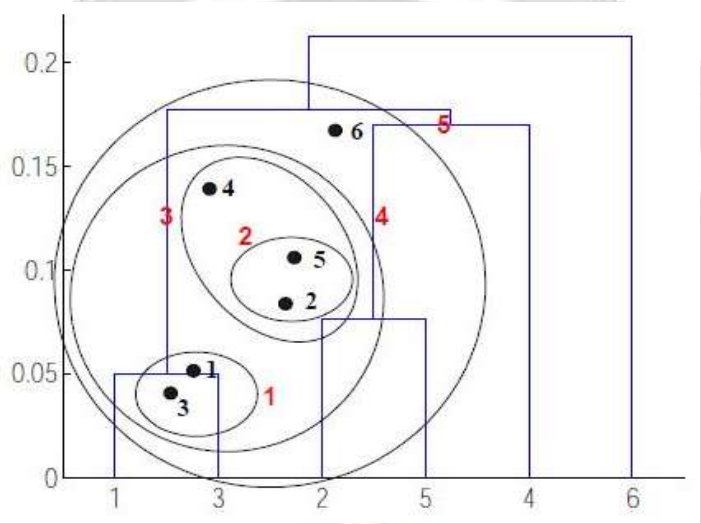

Fig-3: Nested Cluster Diagram. [6]

The clustering found by AGNES can be examined in several different ways of particular interest is the dendrogram, which is a visualization that highlights the kind of exploration enabled by hierarchical clustering over flat approaches such as K-Means. A dendrogram shows data items along one axis and distances along the other axis. The dendrograms in these notes will have the data on the y-axis. The dendrogram shows a collection of shaped paths show the groups that have been joined together. These groups may be the base of another or maybe singleton groups represented as the data along the axis. A key property of the dendrogram is that vertical base which is located along the x-axis according to the distance between the two groups that are being merged. For this to result in a sensible clustering and a valid dendrogram these distances must be monotonically increasing. That is, the distance between two merged groups must always be greater than or equal to the distance between any of the previously-merged subgroups that formed [5].

3.1.1 BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies)

BIRCH is an agglomerative hierarchical based clustering algorithm. It is used for clustering large amounts of data. It is based on the notion of a clustering feature (CF) and a CF tree. A CF tree is a height-balanced tree. Leaf nodes consist of a sequence of clustering features, where each clustering feature represents points that have already been scanned. It is mainly used when a small number of I/O operations are needed. BIRCH uses a multi clustering technique, wherein a basic and good clustering is produced as a result of the first scan, and additional scans can be used to further improve the quality of clustering [1]. The time complexity of BIRCH is O(n) where n is number of clusters [1].

Thus, the agglomerative clustering algorithm will result in a binary cluster tree with single article clusters as its leaf node and a root node containing all the articles. [6]

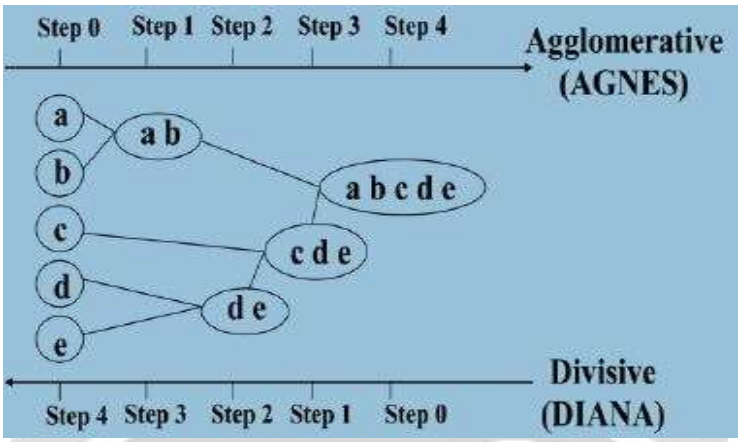

Fig-4: Representation of AGNES and DIANA. [9]

3.2 Divisive (DIANA) Algorithm

This is a "top down" approach. All the observations start with one cluster, and splits are performed recursively as one moves down the hierarchy.

1. Put all objects in one cluster
2. Repeat until all clusters are singletons
   i) Choose a cluster to split
   ii) Replace the chosen cluster with the sub-cluster [6]

3.2.1 Advantages of hierarchical clustering algorithm [9]

1. Embedded flexibility regarding level of granularity.
2. Ease of handling of any forms of similarity or distance.
3. Applicable for any attributes types.

3.2.2 Disadvantages of hierarchical clustering algorithm

1. The fact that most hierarchical algorithms not revisits once constructed clusters for their improvement.
2. Vagueness of termination criteria

4. PARTITION BASED CLUSTERING

4.1 K-Means Clustering Algorithm

K-Means is based on the minimizing the average squared Euclidean distance between the data objects and the cluster's center (called centroids). The results of the algorithm are affected by the initial centroids. Different final clusters produced due to different initial configurations. The center of cluster is defined as the mean of the items in a cluster. [7]

4.1.1 Algorithmic steps for K-Means clustering

1. Choose k data objects from dataset, representing the cluster centroids.
2. Assign each data object of the entire data set to the cluster having the closest centroids.
3. By averaging the data objects belonging to the cluster, compute new centroids for each cluster,.
4. If at least any one of the centroids has changed, go to step 2, otherwise go to step 5.
5. Output the clusters.

The execution of the K-Means algorithm, it tries to improve the running time at the time dealing with high volumes of data. The implementation of K-Means algorithm consists of successive iterations. Each iteration requires

to visit the entire data set in order to assign data objects to their corresponding cluster. While the end of each iteration, new centroids is computed, so the next iteration will use the new centroids. After a certain number of such iterations, the centroids will keep the same and the algorithm stops [7].

Let X = {x1, x2, x3, …….., xn} be the set of data points and V = {v1,v2,…….,vc} be the set of centers. Select 'c' cluster centers randomly. Then Calculate the distance between each of the data point and cluster centers using the Euclidean distance metric as follows,

$$\text{Dist}_{xy} = \sqrt{\sum_{k=1}^{m}(Xik - Xjk)^2}$$

Data point is assigned to the cluster center whose distance from the cluster center is minimum of all the cluster centers. New cluster center is calculated using:

$$\text{Vi} = \left(\frac{1}{Ci}\right)\sum_{1}^{Ci} xi$$

where, 'Ci' denotes the number of data points in i^{th} cluster. The distance between each data point and new obtained cluster centers is recalculated. If no data point was reassigned then stop, otherwise repeat with assigning data point.[8] The K-mean algorithm is used for the clustering in many applications like tweet stream clustering. The basic K-mean algorithm works only on numeric values and prohibits it to cluster real world data containing categorical values.

4.2 Advantages of K-Means clustering algorithm

1. Robust, fast and easier to understand.
2. Relatively efficient: O(tknd), where n is number of objects, k is number of clusters, d is number of dimension of each object, and t is number of iterations. Normally k, d < n.
3. Gives best result when data set are well separated from each other.

4.3 Disadvantages of K-Means clustering algorithm

1. The algorithm requires a priori specification of the number of cluster centers.
2. Randomly choosing of the cluster center cannot lead us to the fruitful result.
3. This algorithm does not work well for categorical data i.e. it is applicable only when mean is defined.
4. Unable to handle noisy data and outliers. [8]

Table-1: Comparison of the features of the various clustering algorithms. [1]

Algorithm	Scalability and Efficiency	Noise	Shape of cluster	Input data
K-Means	Scalable in processing large datasets.	Sensitive to noise and outliers.	Works well only with clusters of convex shapes	Works only on numerical data.
BIRCH	One of the best algorithms for large databases in terms of running time, space required, quality, number of I/O operations applied. Shows linear scalability with respect to a number of objects.		Performs clustering well only with spherical data.	Works on data of all attributes.
DBSCAN	Does not work well for high dimensional data.	Handles noise effectively.	Good at finding clusters of arbitrary shape.	Works on neighbors list, radius, minimum point in radius.

4. CONCLUSIONS

This paper logically compares the various features of DBSCAN, K-means and BIRCH algorithms. DBSCAN is most widely used density based algorithm. It uses the concept of density reachability and density connectivity. BIRCH is an agglomerative hierarchical based clustering algorithm used for clustering large amounts of data. It is based on the notion of a clustering feature (CF) and a CF tree. DBSCAN and hierarchical clustering does not require one to specify the number of clusters a priori, as opposed to k-means. DBSCAN can find arbitrarily shape of clusters.

Hierarchical algorithm shows more quality as compared to k-mean algorithm. BIRCH can work only with spherical data and DBSCAN does not work well over clusters with different densities.

On the other hand, K-means is one of the simplest unsupervised learning algorithms that solve the well known clustering problem-means need certain number of clusters (assume k clusters) fixed a priori. In K-means the centroids should be placed in a cunning way because of different location causes different result.

As the number of records increase the performance of hierarchical algorithm decreasing and the time for execution increased. Standard hierarchical clustering methods can handle data with numeric and categorical values, while K- means can handle only numerical data. The k-means based methods are efficient for processing large data sets. K- Means algorithm is popular because of all. But in future we need such an algorithm which can work efficiently with categorical data, numerical data and arbitrary shape of clusters.

6. REFERENCES

[1]. Mihika Shah, Sindhu Nair, " A Survey of Data Mining Clustering Algorithms", International Journal of Computer Applications (0975 – 8887) Volume 128 – No.1, October 2015.

[2]. Martin Ester, Hans-Peter Kriegel, Jiirg Sander, Xiaowei Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise". KDD-96 Proceedings.

[3]. Adriano Moreira, Maribel Y. Santos and Sofia Carneiro," Density-based clustering algorithms – DBSCAN and SNN", Version 1.0, 25.07.2005.

[4]. Pavel Berkhin, "Survey of Clustering Data Mining Techniques", Accrue Software, Inc.

[5]. Ryan P. Adams, "Hierarchical Agglomerative Clustering".

[6]. Pradeep Rai, Shubha Singh, "A Survey of Clustering Techniques ", International Journal of Computer Applications (0975 – 8887), Volume 7– No.12, October 2010.

[7]. Cosmin Marian Poteras, Marian Cristian Mih˘aescu, Mihai Mocanu, "An optimized Version of the K-Means Clustering Algorithm", Proceedings of the 2014 Federated Conference on Computer Science and Information Systems pp. 695–699, DOI: 10.15439/2014F258 ACSIS, Vol. 2.

[8]. Archana Singh, Avantika Yadav, Ajay Rana, " K-means with Three different Distance Metrics", International Journal of Computer Applications (0975 – 8887) Volume 67– No.10, April 2013.

[9]. Amandeep Kaur Mann & Navneet Kaur, "Review Paper on Clustering Techniques",Global Journal of Computer Science and Technology Software & Data Engineering, Volume 13 Issue 5 Version 1.0 Year 2013, Global Journals Inc. (USA), Online ISSN: 0975-4172 & Print ISSN: 0975-4350.