Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

Comparative Studies of Various Clustering Techniques and Its Characteristics

M.Sathya Deepa
PG & Research Department of Computer Science, Raja Dhoraisingham Govt. Arts College, Sivagangai.
Email: msdeepa06@gmail.com
Dr.N.Sujatha
Assistant Professor, PG & Research Department of Computer Science, Raja Dhoraisingham Govt. Arts College, Sivagangai.

----------ABSTRACT----------
Discovering knowledge from the mass database is the main objective of the Data Mining. Clustering is the key technique in data mining. A cluster is made up of a number of similar objects grouped together. The clustering is an unsupervised learning. There are many methods to form clusters. The four important methods of clustering are Partitional Clustering, Hierarchical Clustering, Density-Based Clustering and Grid-Based Clustering. In this paper, we discussed these four methods in detail.

Keywords: **Clustering, Density-Based, Fuzzy, Grid-Based, Hierarchical, K-Means, Partitioning, STING, Wavelet.**

1. INTRODUCTION

Data mining [1] refers to excavate information from large amount of data. It can also be termed as "Knowledge Discovery from Data (KDD) which is an essential step in the process of knowledge discovery. Data Mining is performed in four steps. Assemble data, Apply data mining tools on datasets, Interpretation and evaluation of result, Result application [5].

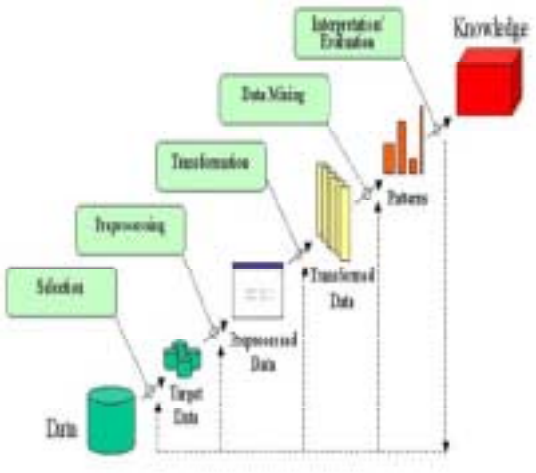

Fig. 1: Steps of Data Mining Process

Data mining practice has the four important jobs. They are Anomaly detection, Association, Classification, Clustering. Anomaly detection is the recognition of odd data records, that may be remarkable or data errors that involve further investigation. Association rule learning is the process to find the relationships between the variables. Classification [2] is the assignment of generalizing the known structure to apply to new data.

In data mining the data is mined using two learning approaches. They are supervised learning and unsupervised learning.

Supervised Learning

Supervised Learning [3] discovers patterns in the data that relate data attributes with a target (class) attribute. These patterns are then utilized to predict the values of the target attribute in future data instances.

UnSupervised Learning

In UnSupervised Learning [35], there is no a priori output. The data have no target attribute. The data is explored to find some intrinsic structures in them.

2. CLUSTERING

Clustering [4] is a significant task in data analysis and data mining applications. The process of organizing the objects into groups whose elements are similar under certain consideration is called Clustering. It is usually performed when no information is available concerning the membership of data items to predefined classes. For this reason, clustering is traditionally seen as part of unsupervised learning. It is useful for the study of inter-relationships among a collection of patterns, by organizing into homogeneous clusters. It is called as unsupervised learning because no a priori labeling of some patterns is available to use in categorize others and infer the cluster structure of the whole data. Intra-connectivity is a measure of the density. A high intra-connectivity is a way to form a good cluster arrangement because the instances grouped within the same cluster are highly dependent on each other. Inter-connectivity is a measure of the

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

connectivity between distinct clusters. A low degree of interconnectivity is advantageous because it indicates that individual clusters are largely independent of each other.

Cluster analysis [6] is a difficult problem because many factors i.e., effective similarity measures, criterion functions, algorithms are come into play in devising a perfect clustering technique for a given clustering problems. Also no clustering method can effectively handle all sorts of cluster structures i.e shape, size and density. Sometimes the quality of the clusters that are found can be improved by preprocessing the given data. It is common to try to find noisy values and eliminate them by a preprocessing step. The input for a system of cluster analysis is a set of samples and a measure of similarity (or dissimilarity) between two samples. The output from cluster analysis is a number of groups /clusters that form a partition, or a structure of partitions, of the data set (Fig.2). The final goal of clustering can be mathematically described as follows:

$$X = C_1 \cup \cdots C_i \cup C_n, \quad C_i \cap C_j = \phi \ (i \neq j),$$

Where *X* denotes the original data set, *Ci*, *Cj* are clusters of X, and n is the number of clusters. Data Pre-processing [7] describes any kind of processing performed on raw data to prepare it for further processing method. This process includes Data cleaning, which fill in missing values, smooth noisy data, identify or remove outliers and resolve in consistencies; Data integration, which integration of multiple data bases, data cubes, or files; Data transformation, which is normalization and aggregation;

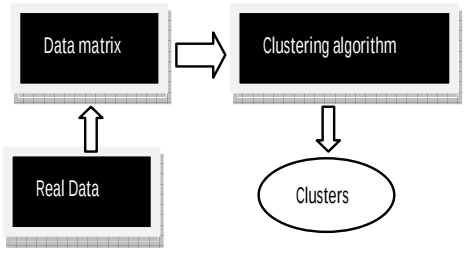

Fig. 2: Clustering Process

Data reduction, which obtains reduced representation in volume but produces the same or similar analytical results. It produces training set.
Clustering has wide applications in

- Image Processing
- Document classification
- Pattern Recognition
- Spatial Data Analysis
- Economic Science
- Cluster Web log data to discover groups of similar access patterns

3. CLASSIFICATION OF CLUSTERS

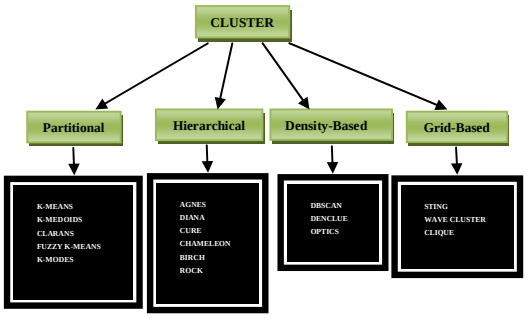

Fig. 3: Classification of Clusters

The Classification of Clusters is shown in the figure 3. They are explained in the following section.

3.1 Partitional Clustering

In Partitional clustering [8], the data points are split into k partition, where each partition represents a cluster. The partition is made based on certain objective function. One such standard function is minimizing the square error. It is calculated by the following formula

$$E = \sum \sum \| p - m_i \|^2$$

where p is the point in a cluster and mi is the mean of the cluster. The cluster must have two properties. They are each group must contain at least one object and each object must belongs to exactly one group. Some of the merits and demerits of Partitional Clustering are

Merits

- More reliable approach.
- Objects may leave one enter another cluster to improve criterion.
- Relatively scalable and simple.
- Suitable for datasets with compact spherical clusters that are well-separated.

Demerits

- Severe effectiveness degradation in high dimensional spaces as almost all pairs of points is about as far away as average.
- The concept of distance between points in high dimensional spaces is ill-defined.
- Poor cluster descriptors.
- Reliance on the user to specify the number of clusters in advance.
- High sensitivity to initialization phase, noise and outliers.
- Frequent entrapments into local optima.
- Inability to deal with non-convex clusters of varying size and density.

In this type of clustering, the familiar algorithms are K-Means, K-Medoids, CLARANS, Fuzzy K-Means, K-Modes. They are explained in the following section.

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

3.1.1 K-Means

K-Means [10] is one of the most popular partitional clustering method in metric spaces. The term "K-Mean" [9] is first proposed by James MacQueen in 1967. But the standard algorithm was firstly introduced by Stuart Lloyd in 1957 as a technique pulse-code modulation. Initially k cluster centroids are selected at random; kmeans then reassigns all the points to their nearest centroids and recomputed centroids of the newly assembled groups. The iterative relocation continues until the criterion function, e.g. square-error converges.

The algorithm steps are

- Choose the number of clusters, *k*.
- Arbitrarily generate *k* clusters and determine the cluster centers, or directly generate *k* random points as cluster centers.
- Assign each point to the nearest cluster center.
- Recompute the new cluster centers.
- Repeat the two previous steps until some convergence criterion is met (usually that the assignment hasn't changed).

Calculate the distance between each object x_i and each cluster center, and then assign each object to the nearest cluster, formula for calculating distance as:

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n}\left\|x_i^{(j)} - c_j\right\|^2 ,$$

Where $\| x_i^{(j)} - c_j \|$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster centre C_j, is an indicator of the distance of the n data points from their respective cluster centres.

$$d(x_i, m_i) = \sqrt{\sum_{j=1}^{d}(x_{i1} - m_{j1})}$$

i = 1,2.....n

j = 1,2.....k

$d(x_i, m_j)$ is the distance between data i and cluster j;

Calculate the mean of objects in each cluster as the new cluster centers,

$$m_i = \frac{1}{N_i}\sum_{j=1}^{N_i} x_{ij}$$

i=l, 2 . . . k; N_i is the number of samples of current cluster i.

3.1.2 K-Medoids

It is also clalled PAM (Partitioning Around Medoids, 1987). The *k*-means algorithm [1] is sensitive to outliers because an object ith an extremely large value may significantly alter the distribution of data. This effect is particularly exacerbated due to the use of the *square*-error function. 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. That is, an absolute-error criterion is used, defined as

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

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 *Cj*; and ***oj*** is the representative object of *Cj*.

The algorithm steps are

- k: the number of clusters, D: a data set containing *n* objects are given.
- Arbitrarily choose *k* objects in *D* as the initial representative objects or seeds.

Repeat

- Assign each remaining object to the cluster with the nearest representative object.
- Randomly select a nonrepresentative object, ***o***random.
- Compute the total cost, *S*, of swapping representative object, ***oj***, with ***o***random.
- If $S < 0$ then swap ***oj*** with ***o***random to form the new set of *k* representative objects;
- Until no change.

3.1.3 CLARANS

Clustering Large Applications based on Randomized Search – CLARANS [11] (Ng and Han, 1994). It combines the sampling techniques with PAM. The clustering process can be presented as searching a graph where every node is a potential solution, that is, a set of k-medoids. The clustering obtained after replacing a medoid is called the neighbor of the current clustering. In this clustering process, certain graph is searched where each node is represented by a set of k medoids in which two nodes are neighbors if they differ by one medoid. Each node has k(T – k) neighbors, where T is the total number of objects.

The computational complexity is

$O((\beta + numlocal)(T - k))$

based on the number of partitions per object or

$O'((\beta + numlocal)k(T - k))$

based on the number of distance calculations, where β is the number of test moves between nodes.

The algorithm steps are

- Randomly choose k mediod

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

- Randomly consider the one of mediod swapped with non mediod
- If the cost of new configuration is lower repeat step 2 with new solution
- If the cost higher repeat step 2 with different non mediod object unless limit has been reached
- Compare the solution keep the best
- Return step 1 unless limit has been reached (set to the value of 2).

3.1.4. Fuzzy K-Means

It is an extension of k-means. Fuzzy K-Means [12] allows data points to be assigned into more than one cluster. Each data point has a degree of membership (or probability) of belonging to each cluster. In fuzzy clustering, each point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely too just one cluster. Thus, points on the edge of a cluster may be in the cluster to a lesser degree than points in the center of cluster. For each point x we have a coefficient giving the degree of being in the kth cluster $u_k(x)$. Usually, the sum of those coefficients is defined to be

$$\forall x \sum_{k=1}^{\text{num. clusters}} u_k(x) = 1.$$

With fuzzy k-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster:

$$\text{center}_k = \frac{\sum_x u_k(x)^m x}{\sum_x u_k(x)^m}.$$

The degree of belonging is related to the inverse of the distance to the cluster center:

$$u_k(x) = \frac{1}{d(\text{center}_k, x)},$$

then the coefficients are normalized and fuzzyfied with a real parameter m > 1 so that their sum is 1. So

$$u_k(x) = \frac{1}{\sum_j \left(\frac{d(\text{center}_k, x)}{d(\text{center}_j, x)} \right)^{2/(m-1)}}.$$

For m equal to 2, this is equivalent to normalizing the coefficient linearly to make their sum 1. When m is close to 1, then cluster center closest to the point is given much more weight than the others, and the algorithm is similar to k-means.

The fuzzy k-means algorithm is very similar to the k-means algorithm.

The algorithm steps are

- Choose a number of clusters.
- Assign randomly to each point coefficients for being in the clusters.
- Repeat until the algorithm has converged (that is, the coefficients' change between two iterations is no more than □, the given sensitivity threshold) .
- Compute the centroid for each cluster, using the formula above.
- For each point, compute its coefficients of being in the clusters, using the formula above.
- The algorithm minimizes intra-cluster variance as well.

3.1.5. K-Modes

The K-means clustering algorithm cannot cluster categorical data because of the dissimilarity measure it uses. The K-modes clustering algorithm [14] is based on K-means paradigm but removes the numeric data limitation while preserving its efficiency. The K-modes algorithm extends K-means paradigm to cluster categorical data by removing the limitation imposed by K-means through following modifications:

- Using a simple matching dissimilarity measure or the hamming distance for categorical data objects
- Replacing means of clusters by their modes.

The algorithm steps are

- Insert the first K objects into K new clusters.
- Calculate the initial K modes for K clusters.
- Repeat {
  - For (each object O)
    - Calculate the similarity between object O and the modes of all clusters.
    - Insert object O into the cluster C whose mode is the least dissimilar to object O.
  - Recalculate the cluster modes so that the cluster similarity between mode and objects is maximized.
  - until (no or few objects change clusters).

Let X, Y be two categorical objects described by m categorical attributes [13]. The simple dissimilarity measure between X and Y is defined by the total mismatches of the corresponding attribute values of the two objects. The smaller the number of mismatches is, the more similar the two objects. Formally,

$$d(X,Y) = \sum_{j=1}^{n} \delta(x_j, y_j)$$

Where

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

$$\delta(x_j, y_j) = \begin{cases} 0 & (x_j = y_j) \\ 1 & (x_j \neq y_j) \end{cases}$$

Let S be a set of categorical objects described by m categorical attributes $A_1,..A_m$. A mode of S = $\{X_1,\ldots X_n\}$ is a vector Q = $[q_1,\ldots q_n]$ that minimizes

$$D(S,Q) = \sum_{i=1}^{n} d(X_i, Q)$$

Here Q is not necessarily an object of S. Let $n_{c_{k,r}}$ be number of objects having the kth category $C_{k,r}$ in attribute A_r and

$$f(A_r = c_{k,r}) = \frac{n_{c_{k,r}}}{n}$$

the relative frequency of category $n_{c_{k,r}}$ in S. The function D(S,Q) is minimized iff
$f(A_r = q_r) \geq f(A_r = C_{k,r})$ for $q_r \neq C_{k,r}$ and all r = 1,…m.
The optimization problem for partitioning a set of n objects described by m categorical attributes into k clusters $S_1,S_2,..S_k$ becomes Minimize

$$\sum_{i=1}^{k} \sum_{X \in S_i} d(X, Q_i)$$

where Qi is the mode of cluster Si.
In the above k-modes clustering problem, the representative point of each cluster Si, i.e., mode Qi is not necessarily contained in Si.

3.2 Hierarchical Clustering

In Hierarchical type of clustering [15], smaller clusters are merged into larger ones, or larger clusters are splitted into smaller clusters. The result of the algorithm is a tree of clusters, called dendrogram, which shows how the clusters are related. By cutting the dendrogram at a desired level, a clustering of the data items into disjoint groups is obtained. A hierarchy of clusters is built by hierarchical clustering. Its representation is a tree, with individual elements at one end and a single cluster with every element at the other (Fig. 4). A hierarchical algorithm yields a dendrogram representing the nested grouping of patterns and similarity levels at which groupings change.

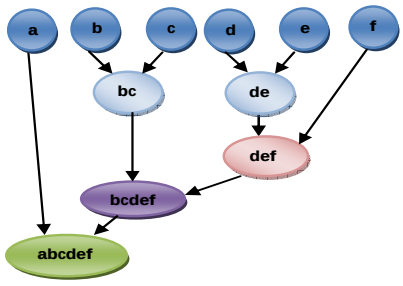

Fig. 4: Hierarchical Clustering

Cutting the tree at a given height will give a clustering at a selected precision. In the above example, cutting after the second row will yield clusters {a} {b c} {d e} {f}. Cutting after the third row will yield clusters {a} {b c} {d e f}, which is a coarser clustering with fewer clusters.

The merging or splitting stops once the desired number of clusters has been formed. In general, 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.

Some of the merits and demerits of the Hierarchical type of Clustering are as follows.

Merits

- Embedded flexibility regarding the level of granularity.
- Ease of handling any forms of similarity or distance.
- Applicability to any attributes type.
- Well suited for problems involving point linkages, e.g. taxonomy trees.

Demerits

- 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.
- Severe effectiveness degradation in high dimensional spaces due to the curse of dimensionality phenomenon.

In Hierarchical Clustering, the familiar algorithms are AGNES, DIANA, CURE, CHAMELEON, BIRCH, ROCK. These are explained in detail in the following section.

3.2.1 AGNES

AGNES – Agglomerative Nesting. It is bottom-up approach. In the beginning [16], every object is a cluster in its own. It first contains *n*-clusters, where *n* is the number of objects in the data set. Algorithms in this category iterate to merge objects which are similar and terminate when there is only one cluster left containing all *n*-data objects. In every stage, method groups the data objects which are most similar.
The algorithm steps [17] are

- Given a set of N objects to be clustered, and an N*N distance (or similarity) matrix, the basic

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

process of hierarchical clustering (defined by S.C. Johnson in 1967) is this:

- Start by assigning each object to a cluster, so that for N objects, we have N clusters, each containing just one object. Let the distances (similarities) between the clusters the same as the distances (similarities) between the objects they contain.
- Find the closest (most similar) pair of clusters and merge them into a single cluster, so that now we have one cluster less.
- Compute distances (similarities) between the new cluster and each of the old clusters.
- Repeat steps 3 and 4 until all items are clustered into a single cluster of size N.
- Step 4 can be done in different ways, and this distinguishes single-linkage, complete linkage and average-linkage clustering
- For Single Linkage: distance between one cluster and another cluster is equal to the shortest distance from any member of one cluster to any member of the other cluster
- For Complete Linkage: distance between one cluster and another cluster to be equal to the greatest distance from any member of one cluster to any member of the other cluster.

From the figure 4, (which represents the hierarchical type of clustering also represents the agglomerative type of clustering [23]) Suppose we have merged the two closest elements *b* and *c*, we now have the following clusters {*a*}, {*b*, *c*}, {*d*}, {*e*} and {*f*}, and want to merge them further. To do that, we need to take the distance between {a} and {b c}, and therefore define the distance between two clusters. Usually the distance between the two clusters **A** and **B** is one of the following:

The maximum distance between elements of each cluster (also called complete-linkage clustering):

$$\max\{\, d(x,y) : x \in \mathcal{A},\, y \in \mathcal{B} \,\}.$$

The minimum distance between elements of each cluster (also called single-linkage clustering):

$$\min\{\, d(x,y) : x \in \mathcal{A},\, y \in \mathcal{B} \,\}.$$

The mean distance between elements of each cluster (also called average linkage clustering):

$$\frac{1}{|\mathcal{A}| \cdot |\mathcal{B}|} \sum_{x \in \mathcal{A}} \sum_{y \in \mathcal{B}} d(x,y).$$

3.2.2 DIANA

DIANA – Divisive Analysis [18]. It is top-down approach. It is introduced in Kaufmann and Rousseeuw (1990). It is the inverse of the agglomerative method. These algorithms produce a sequence of clustering schemes of increasing number of clusters at each step. It produces each step result from the previous one by splitting a cluster into two.

The algorithm steps [19] are

- It starts with a single cluster, the entire set of n objects.
- In each step, the cluster with largest diameter is selected and is to be divided into two clusters. Here the diameter of a cluster is defined as the maximum distance or dissimilarity (i.e., minimum similarity) among all objects within the cluster.
- An object within this cluster having largest average dissimilarity to other objects within the cluster is identified. This object initiates 'splinter group.'
- An object within this cluster is reassigned to the splinter group if it is closer to the splinter group than to the 'old party.'
- At the end of the step, the cluster is divided into two new clusters.
- The above step is repeated until n clusters are formed.

The dendrogram produced by the method Diana is given in the following figure – 5.

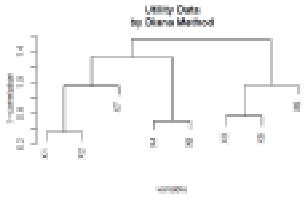

Fig. 5: Dendrogram of Diana

3.2.3 CURE

CURE - Clustering Using Representatives [20] - A hierarchical clustering algorithm for large databases. It is proposed by Guha, Rastogi and Shim. This algorithm has 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. For each iteration, the pair of clusters with the closest representatives is merged.

The algorithm steps are

- Set a target representative points number ***c***, for each cluster, select ***c*** *well scattered* points attempting to capture the physical shape and geometry of the cluster.
- The chosen scattered points are then shrunk toward the centroid in a fraction of α where $0 \leq \alpha \leq 1$.

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

- These points are used as representatives and at each step of the algorithm, two clusters with closest pair of representatives are merged (d_{min}).
- After each merging, another c points are selected from original representatives of previous clusters to represent new cluster.
- Cluster merging stops until target k clusters are found.

3.2.4 CHAMELEON

It [21] is a newly developed agglomerative Hierarchical Clustering algorithm based on the nearest-neighbor graph, in which an edge is eliminated if both vertices are not within the closest points related to each other. It [7] can find dynamic modeling. It is based on two phases:

- Use a graph partitioning algorithm: cluster objects into a large number of relatively small sub-clusters.
- Use an agglomerative hierarchical clustering algorithm: find the genuine clusters by repeatedly combining these sub-clusters.

The algorithm steps are

- At the first step, the connectivity graph is divided into a set of subclusters with the minimal edge cut.
- Each subgraph should contain enough nodes in order for effective similarity computation. By combining the relative interconnectivity and relative closeness make to explore the characteristics of potential clusters.
- These small subsets are merged and thus we get the ultimate clustering solutions.

Here, the relative interconnectivity (or closeness) is obtained by normalizing the sum of weights (or average weight) of the edges connecting the two clusters over the internal connectivity (or closeness) of the clusters. Fig 6 provides an overview of the overall approach used by Chameleon to find the clusters in a data point.

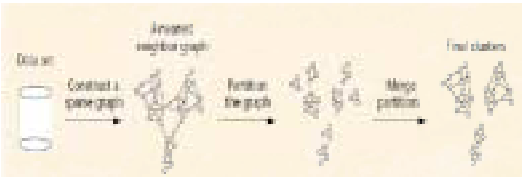

Fig. 6: CHAMELEON Process

3.2.5 BIRCH

BIRCH - Balanced Iterative Reducing and Clustering using Hierarchies. This method [22] has been designed so as to minimize the number of I/O operations. It is incrementally and dynamically forms clusters using incoming multi-dimensional metric data points to try to produce the best quality clustering with the available memory and time constraints. It is the first clustering algorithm planned in the database area to handle "noise". This method introduces two concepts, clustering feature and clustering feature tree (CF tree), which are used to review cluster representations. These structures help the clustering method achieve good speed and scalability in large databases.

Incrementally construct a CF (Clustering Feature) tree, a hierarchical data structure for multiphase clustering.

Phase 1: scan DB to build an initial in-memory CF tree (a multi-level compression of the data that tries to preserve the inherent clustering structure of the data)

Phase 2: use an arbitrary clustering algorithm to cluster the leaf nodes of the CF-tree.

Given n d-dimensional data objects or points in a cluster, we can define the centroid $\boldsymbol{x_0}$, radius R, and diameter D of the cluster as follows:

$$x_0 = \frac{\sum_{i=1}^{n} x_i}{n}$$

$$R = \sqrt{\frac{\sum_{i=1}^{n} (x_i - x_0)^2}{n}}$$

$$D = \sqrt{\frac{\sum_{i=1}^{n} \sum_{j=1}^{n} (x_i - x_j)^2}{n(n-1)}}$$

where R is the average distance from member objects to the centroid, and D is the average pairwise distance within a cluster. Both R and D reflect the tightness of the cluster around the centroid.

Clustering Feature

A clustering feature (CF) is a three-dimensional vector summarizing information about clusters of objects. Given n d-dimensional objects or points in a cluster, $\{x_i\}$, then the CF of the cluster is defined as

$$CF = \langle n, LS, SS \rangle,$$

where n is the number of points in the cluster, $\boldsymbol{LS}$ is the linear sum of the n points , SS is the square sum of data points.

Clustering Feature Tree

- Each non-leaf node has at most B entries.
- Each leaf node has at most L CF entries, each of which satisfies threshold T.
- Node size is determined by dimensionality of data space and input parameter P (page size).

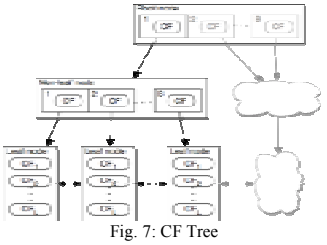

Fig. 7: CF Tree

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

Clustering features are sufficient for calculating all of the measurements that are needed for making clustering decisions in BIRCH. BIRCH thus utilizes storage efficiently by employing the clustering features to summarize information about the clusters of objects, thereby bypassing the need to store all objects.

3.2.6 ROCK

ROCK - RObust Clustering using links. It [24][25] performs agglomerative hierarchical clustering and explores the concept of links for data with categorical attributes.
The various attributes are defined below:-

Links - The number of common neighbours between two objects.

Neighbors - If similarity between two points exceeds certain similarity threshold (θ), they are neighbours i.e., if similarity (A,B)≥θ then only two points A, B are neighbours, where similarity is a similarity function and θ is a user-specified threshold.

Criterion Function - The objective is to maximize the criterion function to get the good quality clusters. By maximizing we mean maximizing the sum of links of intra cluster point pairs while minimizing the sum of links of inter cluster point pairs.

$$E_l = \sum_{i=1}^{k} n_i \times \sum_{p_q, p_r \in C_i} \frac{link(p_q, p_r)}{n_i^{1+2f(\theta)}}$$

Goodness Measure - While performing clustering the motive of using goodness measure is – to maximize the criterion function and to identify the best pair of clusters to b merged at each step of ROCK.

$$g(C_i, C_j) = \frac{link[C_i, C_j]}{(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}}$$

Where C_i,C_j - pair of clusters, link[C_i,C_j] –number of number links between clusters C_i and C_j, g(C_i,C_j) – goodness measure.

Jaccard's coefficient: Jaccard's coefficient is a good similarity measure because it can find the similarity between the categorical data.
For sets A and B of keywords used in the documents, the Jaccard coefficient [4] may be defined as follows:

Similarity (A, B) = (|A ∩ B|) / (A U B|)

The algorithm steps are

- A sample set of documents, Number of k clusters to be found, the similarity threshold for this task: θ≥0.4 are given as input.
- Take k and θ≥0.4
- Initially, place each document into a separate cluster.
- Construction of Similarity Matrix: Constructing the similarity matrix by computing similarity for each pair of queries (A,B) using measure for instance i.e.
  Similarity (A, B) = (|A ∩ B|)/(A U B|)
- Computation of Adjacency Matrix : Compute Adjacency Matrix (A) using similarity threshold θ≥0.4 i.e.
  if similarity(A, B)≥θ then 1; else 0
- Computation of Links: Compute Link Matrix by multiplying Adjacency Matrix to itself i.e. A x A to find the number of links.
- Calculation of Goodness Measure: The goodness measure for each pair of documents is calculated by using the following function:

$$g(C_i, C_j) = \frac{link[C_i, C_j]}{(n_i + n_j)^{1+2f(\theta)} - n_i^{1+2f(\theta)} - n_j^{1+2f(\theta)}}$$

Where f (θ) = (1-θ)/(1+θ).

- Merge the two documents with the highest similarity (goodness measure).
- When no more entry exists in the goodness measure table then stop algorithm by resulting in k number of clusters and noise (if any) otherwise go to step 4.
- A group of documents i.e. clusters are the outputs we get.

3.3 Density-Based Clustering

Density-Based Clusters [26] are defined as areas of higher density than the remainder of the data set. Objects in these sparse areas - that are required to separate clusters - are usually considered to be noise and border points. It requires just two parameters and is mostly in sensitive to the ordering of the database. The quality of density-based clustering depends on the distance measure used in the function. It does not require one to specify the number of clusters in the data a priori.

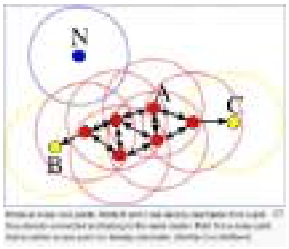

Fig. 8: Density-Based Clustering

The aim of these methods is to identify the clusters and their distribution parameters. These methods [27] are designed for discovering clusters of arbitrary shape which are not necessarily convex.

$$x_i, x_j \in C_k$$

This does not necessarily imply that:

$$\alpha \cdot x_i + (1 - \alpha) \cdot x_j \in C_k$$

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

Where C_k is the Cluster ,k is the number of clusters and x_i and $x_j$are the distribution parameters and α is the threshold value.

Merits and Demerits of Density-based Clustering are

Merits

- Discovery of arbitrary-shaped clusters with varying size
- Resistance to noise and outliers.

Demerits

- High sensitivity to the setting of input parameters
- Poor cluster descriptors
- Unsuitable for high-dimensional datasets because of the curse of dimensionality phenomenon.

The algorithms in this method include DBSCAN, DENCLUE and OPTICS. These are explained in the following sections.

3.3.1 DBSCAN

The density based algorithm DBSCAN [28] - Density Based Spatial Clustering of Applications with Noise. It is one of the most common clustering algorithms and also most cited in scientific literature. Unlike K-Means, number of clusters not needed before running algorithm. It works well with convex clusters. Heuristic methods are used for picking parameters.

Core object - object (***p***) that has a minimum number of neighbors within a certain radius.
Border object - object (***q***), not ***p***, but a neighbor of ***p***
Noise - neither ***p*** nor ***q***.
Directly density-reachable - ***q*** is *directly density-reachable* from ***p***, if ***q*** is a neighbor of ***p***.
Density-reachable - ***q*** is *density reachable* from ***p***, if ***p***->***p1***, ***q***->***pn***, and for every ***pi+1*** is directly density reachable from ***pi***.
Density-connected - ***p*** and ***q*** are density connected if there is an object, ***o***, for which ***o*** is density reachable from ***p*** and ***q***.
The Eps and the Minpts are the two parameters of the DBSCAN. The basic idea of DBSCAN algorithm is that a neighborhood around a point of a given radius (ε) must contain at least minimum number of points (MinPts) .
The algorithm steps are:

- Randomly select a point t
- Recover all density-reachable points from t wrt Eps and MinPts.
- Cluster is created, if t is a core point.
- If t is a border point, no points are density-reachable from t and DBSCAN visits the next point of the database.
- Continue the procedure until all of the points have been processed.

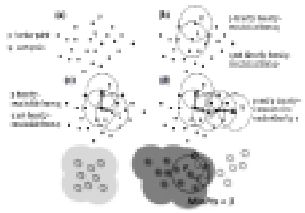

Fig. 9: DBSCAN

3.3.2 DENCLUE

DENCLUE [29] - DENsity-based CLUstEring developed by Hinneburg & Keim. It is a clustering method based on density distribution functions. This algorithm is built on the following ideas:

- The influence of each data point can be formally modeled using a mathematical function called an influence function.
- The overall density of the data space is the sum of the influence function applied to all data points.
- Clusters can then be determined mathematically by identifying density attractors, where density attractors are local maxima of the overall density function.

Influence function
Let x and y be objects or points in F^d, a d-dimensional input space.
The influence function of data object y on x is a function:

$$f_B^y(x) = f_B(x, y)$$

It can be used to compute a square wave influence function,

$$f_{Square}(x,y) = \begin{cases} 0 & if\ d(x,y) > \sigma \\ 1 & otherwise, \end{cases}$$

or a Gaussian influence function,

$$f_{Gauss}(x,y) = e^{-\frac{d(x,y)^2}{2\sigma^2}}$$

σ is a threshold parameter.

Density function

- The density function at an object or point *x* is defined as the sum of influence functions of all data points. That is, it is the total influence on ***x*** of all of the data points.
- Given *n* data objects, the density function at *x* is defined as

$$f_B^D(x) = \sum_{i=1}^{n} f_B^{x_i}(x) = f_B^{x_1}(x) + f_B^{x_2}(x) + \cdots + f_B^{x_n}(x)$$

The algorithm steps [30] are

- Take Data set in Grid whose each side is of 2σ.
- Find highly densed cells.
- Find out the mean of highly populated cells

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

- If d(mean(c_1), mean(c_2)) < 4a then two cubes are connected.
- Now highly populated or cubes that are connected to highly populated cells will be considered in determining clusters.
- Find Density Attractors using a Hill Climbing Procedure.
- Randomly pick point r.
- Compute Local 4 σ density
- Pick another point (r+1) close to previous computed density.
- If den(r) < den(r+1) climb.
- Put points within (σ /2) of path into cluster.
- Connect the density attractor based cluster.

Figure 10 shows a 2-D data set together with the corresponding overall density functions for a square wave and a Gaussian influence function.

Fig. 10: Possible density functions for a 2-D data set.

3.3.3 OPTICS

OPTICS [31] - Ordering Points to Identify the Clustering Structure. It is similar to DBSCAN but produces augmented cluster ordering instead of defining actual clusters. This approach is advantageous as the ordering can be used to both derive key cluster characteristics and analyse the structure of the cluster space, and time complexity is the same as DBSCAN; averaging O(n log n).

The OPTICS technique builds upon DBSCAN by introducing values that are stored with each data object; an attempt to overcome the need to supply different input parameters. In particular, these are referred to as the core distance, the smallest epsilon value that makes a data object a core object, and the reachability-distance, which is a measure of distance between a given object and another. The reachability-distance is calculated as the greater of either the core-distance of the data object or the Euclidean distance between the data object and another point. These newly introduced distances are used to order the objects within the data set. Clusters are defined based upon the reachability information and core distances associated with each object; potentially revealing more relevant information about the attributes of each cluster.

Core-distance of an object p

Let p [32] be an object from a database D, let □ be a distance value, let $N_{□}(p)$ be the □ -neighborhood of p, let MinPts be a natural number and let MinPts-distance(p) be the distance from p to its MinPts' neighbor. Then, the core-distance of p is defined as Core-Distance$_{□,MinPts}$(p) =

$$\begin{cases} UNDEFINED, \text{ if } Card(N_{□}(p)) < MinPts \\ MinPts\text{-}distance(p), \text{ otherwise} \end{cases}$$

The core-distance of an object p is simply the smallest distance □' between p and an object in its □-neighborhood such that p would be a core object with respect to □' if this neighbor is contained in $N_{□}(p)$. Otherwise, the core-distance is UNDEFINED.

reachability-distance object p w.r.t. object *o*

Let p and o be objects from a database D, let $N_{□}(o)$ be the □-neighborhood of *o*, and let MinPts be a natural number. Then, the reachability-distance of p with respect to *o* is defined as

Reachability-distance$_{□, MinPts}$ (p, *o*) =

$$\begin{cases} UNDEFINED, \text{ if } |N_{\varepsilon}(o)| < MinPts \\ max(core\text{-}distance(o), distance(o, p)), \text{otherwise} \end{cases}$$

Naturally, the reachability-distance of an object p with respect to another object *o* is the smallest distance such thatp is directly density-reachable from *o* if *o* is a core object. Otherwise, if *o* is not a core object, even at the generating distance □, the reachability-distance of p with respect to *o* is UNDEFINED. The reachability-distance of an object p depends on the core object with respect to which it is calculated. Figure 11 illustrates the notions of core-distance and reachability-distance.

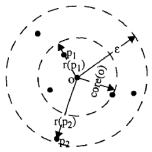

Fig. 11: Core-distance(*o*), reachability-distances $r(p_1,o)$,$r(p_2,o)$ for MinPts=4

The algorithm steps are

- Create an ordering of the objects in a database, storing the core-distance and a suitable reachability-distance for each object. Clusters with highest density will be finished first.
- Based on the ordering information produced by OPTICS, use another algorithm to extract clusters.
- Extract density-based clusters with respect to any distance e' that is smaller than the distance e used in generating the order.

3.4 Grid-Based Clustering

The Grid-Based type of clustering approach [33] uses a multi resolution grid data structure. It quantizes the object space into a finite number of cells that form a grid structure on which all of the operations for clustering are performed. The grid-based clustering approach differs from the conventional clustering algorithms in that it is concerned not with the data

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

points but with the value space that surrounds the data points. In general, a typical grid-based clustering algorithm consists of the following five basic steps:

- Creating the grid structure, i.e., partitioning the data space into a finite number of cells.
- Calculating the cell density for each cell.
- Sorting of the cells according to their densities.
- Identifying cluster centers.
- Traversal of neighbor cells.

The Merits and Demerits are as follows

Merits

- Fast processing time.
- Good cluster quality.
- No distance computations
- Clustering is performed on summaries and not individual objects; complexity is usually O(#-populated-grid-cells) and not O(#objects)
- Easy to determine which clusters are neighboring
- Shapes are limited to union of grid-cells

Demerits

- All the cluster boundaries are either horizontal or vertical, and no diagonal boundary is detected.

The important algorithms in this method include STING, Wavelet and CLIQUE. These are explained in the following sections.

3.4.1 STING

STING [6] - STatistical INformation Grid. It is a grid based multi resolution clustering technique in which the spatial area is separated into rectangular cells (using latitude and longitude) and employs a hierarchical structure. Several levels of such rectangular cells represent different levels of resolution. Figure 12 shows the hierarchical structure of STING Clustering.

The algorithm steps are

- At lower level, each cell is partitioned in to child cells
- A cell in level 'i' corresponds to union of its children at level i + 1.
- Each cell (except the leaves) has 4 children & each child corresponds to one quadrant of the parent cell.
- Statistical information regarding the attributes in each grid cell is pre computed and stored.
- Statistical parameters of higher level cells can easily be computed from the parameters of lower level cells.
- For each cell, there are attribute independent parameters and attribute dependant parameters. (Attribute independent parameter - count and Attribute dependant parameters - M: Mean of all values in the cell; S: Standard deviation of all values in this cell Min: minimum value of the value of the attribute in this cell).
- The distribution types - normal, uniform, exponential or none.
- Value of distribution may either be assigned by the user or obtained by hypothesis tests such as X_2 test.
- When data are loaded into database, parameters calculate count,m,s, min,max of the bottom level cells directly.
- Irrelevant cells are removed and this process is repeated until the bottom layer is reached.

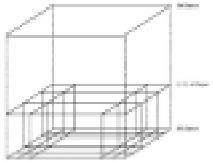

Fig. 12: A Hierarchical Structure of STING Clustering

3.4.2 Wave Cluster

Wave Cluster [34] is a multi-resolution clustering approach which applies wavelet transform to the feature space. A wavelet transform is a signal processing technique that decomposes a signal into different frequency sub-band. . Given a set of spatial objects $\mathbf{o_i}$, $1 \leq i \leq N$, the goal of the algorithm is to detect clusters and assign labels to the objects based on the cluster that they belong to. The main idea in WaveCluster is to transform the original feature space by applying wavelet transform and then find the dense regions in the new space. It yields set of clusters at different resolutions and scales, which can be chosen based on the user's needs.

The algorithm steps are

- Multidimensional data objects' feature vectors are given as input.
- The feature space is quantized then objects are assigned to the cells.
- Wavelet transform is applied on the quantized feature space.
- At different levels, the connected components (clusters) in the subbands of transformed feature space are found.
- Labels are assigned to the cells.
- The lookup table is created.
- The objects to the clusters are plotted.

Fig. 13: A Sample of two dimensional feature space.

Figure 13 shows a sample of two dimensional feature space, where each point in the image represents the

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

attribute or feature values of one object in the spatial data set.

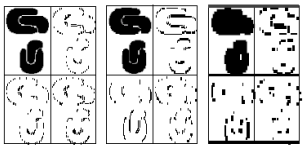

Fig. 14: Multiresolution of the feature space in Figure 13 at (a) scale 1 (high resolution); (b) scale 2 (medium resolution); and (c) scale 3 (low resolution).

Figure 14 shows the resulting wavelet transformation at different resolutions, from a fine scale (scale 1) to a coarse scale (scale 3).

3.4.3 CLIQUE

CLIQUE [22] - CLustering In QUEst. It can be considered as both density-based and grid-based.

The algorithm steps are

- It partitions each dimension into the same number of equal length interval.
- It partitions an m-dimensional data space into non-overlapping rectangular units.
- A unit is dense if the fraction of total data points contained in the unit exceeds the input model parameter
- A cluster is a maximal set of connected dense units within a subspace.

It is *insensitive* to the order of records in input and does not presume some canonical data distribution. It scales *linearly* with the size of input and has good scalability as the number of dimensions in the data increases.

Characteristics of Various Clustering Algorithms

ALGORITHMS	DATASET	DATA TYPE	MEASURES	PRIMARY DATA REQUIRED	COMPLEXITY	CLUSTER SHAPE
K-MEANS	Large	Numerical	Mean	Number of Clusters	o(nkl)	Spherical
K-MEDOIDS	Large	Numerical	Medoid	Number of Clusters	o(nkl)	Arbitrary
CLARANS	Sample	Numerical	Medoid	Number of Clusters	$o(n^2)$	Arbitrary
FUZZY K-MEANS	Large	Numerical	Measure	Number of Clusters	o(tkn)	Arbitrary
K-MODES	Large	Mixed	Mode	Number of Clusters	o(tkn)	Spherical
AGNES	Large	Numerical	Similarity Measure	Number of Clusters	$o(n^2)$	Tree
DIANA	Large	Numerical	Similarity Measure	Number of Clusters	$o(2^n)$	Tree
CURE	Large	Numerical	Similarity Measure	Number of Clusters	$o(n^2 \log n)$	Arbitrary
CHAMELEON	Sample	Discrete	Similarity Measure	Minimum Similarity	$o(n^2)$	Arbitrary
BIRCH	Large	Numerical	Feature Tree	Number of Clusters	o(n)	Spherical
ROCK	Small Sized	Mixed	Similarity Measure	Number of Clusters	$o(kn^2)$	Graph
DBSCAN	High Dimensional	Numerical	Density Based	Density Threshold	o(nlogn)	Arbitrary
DENCLUE	High Dimensional	Numerical	Density Based	Radius	$o(n^2)$	Arbitrary
OPTICS	High Dimensional	Numerical	Density Based	Density Threshold	o(nlogn)	Arbitrary
STING	Any Size	Numerical	Statistical	Statistical	o(n)	Rectangular
WAVE CLUSTER	Low Dimensional	Numerical	Wave Transform	Wavelet Transform	o(n)	Arbitrary
CLIQUE	High Dimensional	Mixed	Density Based	Density Threshold	o(n)	Arbitrary

Table 1: Characteristics of various clustering algorithms

4. CONCLUSION

Discovering knowledge from the mass database is the main objective of the Data Mining. Clustering is the key technique in data mining. A cluster is made up of a number of similar objects grouped together. This paper gives the review of four important techniques namely Partitional Clustering, Hierarchical Clustering, Density Based Clustering and Grid Based Clustering. The different algorithms of these techniques are discussed. So this paper provides a quick review of these four clustering techniques.

References

[1]. Jiawei Han and M Kamber , Data Mining: Concepts and Techniques, Second Edition.

[2]. Amandeep Kaur Mann, “Review paper on Clustering Techniques”, Global Journal of Computer Science and Technology Software & Data Engineering.

[3]. “Unsupervised Learning”, 22c:145 Artificial Intelligence, The University of Iowa.

[4]. Nizar Grira, Michel Crucianu, Nozha Boujemaa, “Unsupervised and Semi-supervised Clustering: a Brief Survey”, INRIA Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex, France.

[5]. Navneet Kaur, “Survey Paper on Clustering Techniques”, International Journal of Science, Engineering and Technology Research (IJSETR).

[6]. B.G.Obula Reddy, Dr. Maligela Ussenaiah, “Literature Survey on Clustering Techniques”, IOSR Journal of Computer Engineering (IOSRJCE).

[7]. M. Kuchaki Rafsanjani, Z. Asghari Varzaneh, N. Emami Chukanlo , “A survey of hierarchical

Volume: 5 Issue: 6 Pages: 2104-2116 (2014) ISSN : 0975-0290

clustering algorithms", TJMCS Vol .5 No.3 (2012) 229-240 .
[8]. S. Anitha Elavarasi, Dr. J. Akilandeswari,Dr. B. Sathiyabhama, "A SURVEY ON PARTITION CLUSTERING ALGORITHMS", International Journal of Enterprise Computing and Business Systems.
[9]. Pritesh Vora, Bhavesh Oza, "A Survey on K-mean Clustering and Particle Swarm Optimization", International Journal of Science and Modern Engineering (IJISME).
[10]. Deepti Sisodia, Lokesh Singh, "Clustering Techniques: A Brief Survey of Different Clustering Algorithms", International Journal of Latest Trends in Engineering and Technology (IJLTET).
[11]. Shruti Aggrwal and Prabhdip Kaur, " Survey of Partition Based Clustering Algorithm Used for Outlier Detection", INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY.
[12]. Ramandeep Kaur & Gurjith Singh Bhathal, "A Survey of Clustering Techniques", International Journal on Computer Science and Engineering Vol. 02, No. 09, 2010, 2976-2980.
[13]. Zengyou He, "Approximation Algorithms for K-Modes Clustering".
[14]. Shehroz S Khan and Dr. Shri Kant, "Computation of Initial Modes for K-modes Clustering Algorithm using Evidence Accumulation", IJCAI-07 ,2784.
[15]. Literature Survey: Data Clustering Algorithms and Soft Computing,Chapter 2.
[16]. Swati Harkanth, Prof. B. D. Phulpagar, "A Survey on Clustering Methods and Algorithms", International Journal of Computer Science and Information Technologies, Vol. 4 (5) , 2013, 687-691.
[17]. Gagandeep Kaur, "Clustering".
[18]. Maria Halkidi, Yannis Batistakis, Michalis Vazirgiannis , "On Clustering Validation Techniques", Journal of Intelligent Information Systems.
[19]. http://www.stat.wmich.edu/wang/561/classnotes/Grouping/Cluster.pdf
[20]. Xuanqing Chu and Zhuofu Song, "CURE: An Efficient Clustering Algorithm for Large Databases ".
[21]. Rui Xu, and Donald Wunsch II , "Survey of Clustering Algorithms", IEEE TRANSACTIONS ON NEURAL NETWORKS, VOL. 16, NO. 3, MAY 2005.
[22]. Bill Andreopoulos , "Literature Survey of Clustering Algorithms".
[23]. Hierarchical Clustering- Wikipedia.
[24]. Ashwina Tyagi and Sheetal Sharma, "Implementation Of ROCK Clustering Algorithm For The Optimization Of Query Searching Time", International Journal on Computer Science and Engineering.
[25]. Sudipto Guha, Rajeev Rastogi, Kyuseok Shim , "ROCK: A robust clustering algorithm for categorical attributes".
[26]. Anoop Kumar Jain & Prof. Satyam Maheswari, "Survey of Recent Clustering Techniques in Data Mining", International Journal of Computer Science and Management Research
[27]. Lior Rokach & Oded Maimon, "CLUSTERING METHODS".
[28]. Parallel DBSCAN - Oliver Sampson
[29]. Dr. Masoud Yaghini, "Cluster Analysis--Density-Based Methods", Spring 2010.
[30]. Pooja Batra Nagpal and Priyanka Ahlawat Mann, "Comparative Study of Density based Clustering Algorithms", International Journal of Computer Applications (0975 – 8887) Volume 27– No.11.
[31]. David Breitkreutz and Kate Casey, "Clusters: a Comparison of Partitioning and Density-Based Algorithms and a Discussion of Optimizations".
[32]. [Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, J&g Sander, "OPTICS: Ordering Points To Identify the Clustering Structure".
[33]. Wei Cheng, Wei Wang and Sandra Batista, "Data Clustering: Algorithms and Applications", Chapter 1.
[34]. Gholamhosein Sheikholeslami, Surojit Chatterjee, Aidong Zhang, "WaveCluster: a wavelet-based clustering approach for spatial data in very large databases", The VLDB Journal (2000) 8: 289–304.
[35]. http://sisla06.samsi.info/jpal/mult1031.pdf