ISSN: 2277-9655
Impact Factor: 1.852

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

A Comparative Study on Hierarchical Clustering in Data Mining

K Kiruba[*1], Dr B Rosiline Jeetha[2]

[*1] Research Scholar, Department of Computer Science, RVS College of Arts & Science, Coimbatore,
Tamil Nadu-641402, India

[2]Associate Professor, Department of Computer Applications (MCA), RVS College of Arts & Science,
Coimbatore, Tamil Nadu-641402, India

kiruba2090@gmail.com

Abstract

Data mining is largely concerned with building models. Model is simply an algorithm or set of rules that connects a collection of data (input) to a particular target or outcome. Data mining involves the tasks are classification, estimation, prediction, clustering, affinity grouping, description & profiling. The first 3 are all the examples of directed data mining, where the goal is to find the value of a particular target variable. Affinity grouping and clustering are undirected tasks where the goal is to uncover structure in data without respect to particular target variable.

Profiling in a descriptive task that may be either directed or undirected. In this paper we will review the main methods and approaches of clustering. Clustering is the task of segmenting a heterogeneous population into a number of more homogeneous sub groups or clusters. This survey concentrated on data mining, data mining issues, clusters, clustering, clustering analysis, clustering algorithms, clustering issues, comparison of clustering algorithm, and Requirements of clustering in data mining.

Keywords: Data mining, Clustering, Hierarchical clustering algorithm, Agglomerative, Divisive.

Introduction

Data mining is largely concerned with building models. Model is simply an algorithm or set of rules that connects a collection of data to a particular target or outcome. Classification, estimation, prediction, clustering, affinity grouping, description and profiling are the tasks involved in data mining. The first three are the examples of directed data mining, where the goal is to find the value of a particular target variable. Affinity grouping and clustering are un-directed tasks where the goal is to uncover structure in data without respect to a particular target variable. Profiling in a descriptive task that may be either directed or un-directed [1].

Clustering is the task of segmenting a heterogeneous population into a number of more homogeneous sub groups or clusters. Clustering and clusters are not synonymous. A clustering is an entire collection of clusters; a cluster on the other hand is just one part of the entire picture. There are different types of clusters and also different types of clustering.

Cluster Analysis

Cluster analysis is the process of grouping data objects based on the information present, the grouping criteria is that objects within a cluster or group be similar to one another and different from objects in another group. Clustering is different from the previously discussed concept of classification (chapter 1) in that clustering does not assign labels that have been previously determined to the groups; it only separates the data objects into groups. Any labeling that may occurs does not depend on previously classified data objects that we have access to.

What is Cluster Analysis?

Cluster analysis aims to group data objects based on the information that is available that describes the objects and their relationships. The main goal is to group similar objects together, and the greater the similarity within a group the better and the greater the difference between group the more diverse the clustering. Clustering is a form of unsupervised learning because as previously mentioned we do not have a data set has been previously labeled to train the current data on. For instance, a set of data points while outlines possible clusters that may exist. If this was supervised learning we would have a dataset that has already been classified that can be used to control and guide the process.

Understanding

When it comes to data analysis for the purpose of understanding the dataset, cluster analysis is the study of techniques for automatically finding classes because every cluster is a potential class just needed a class label. Applications for this use of clustering exist in the fields of biology when it comes to taxonomy and grouping genetic information, information retrieval, climate to help find patterns in the atmosphere and ocean. In the field of psychology and medicine, clustering is used for diagnosis of diseases and in business it is used to segment customers into small groups that can later be targeted for future marketing activities.

Utility

Cluster analysis can also be used as the basis for other data analysis or processing techniques, in this context, cluster analysis is similar to visualization it is the study of techniques for finding the most representative clusters. Applications for this use of clustering include summarization which uses clustering to avoid the curse of dimensionality and apply the algorithm to cluster prototypes. Clustering can also be used to efficiently find nearest neighbours.

.

Types of Clusters

Clusters can be created based on varying characteristics, some of which are mentioned briefly below.

Well-Separated: Clusters can be well-separated by this we mean that the distance between any two points in different clusters is large and the distance between any two objects in a cluster is relatively smaller. **Prototype-Based:** A cluster is defined based on the prototype by this we mean that each object is classified by its proximity to a prototype. **Graph-Based:** In situations where the data is represented by a graph and the nodes are objects with connection visualized by links a cluster is a group of objects that are connected to each other but have no connection to objects outside of the group. **Density-Based:** This type of cluster is defined by the density of the region. A cluster is a dense region of objects surrendered by a region of low density.

Types of Clustering Algorithms

- **Partitioning-based clustering:** are algorithms that determine all the clusters at once in most cases.
  - K-means clustering
  - K-medoids clustering
  - EM (expectation maximization) clustering
- **Density-Based Methods**: these clustering algorithms are used to help discover arbitrary-shaped clusters. A cluster is defined as a region in which the density of data objects exceeds some threshold.
  - DBSCAN
  - OPTICS
- **Hierarchical clustering:** these algorithms find successive clusters using previously established ones.
  - Agglomerative clustering(Bottom up approach)
  - Divisive clustering(Top down approach)

Hierarchical Clustering

Partitioning algorithms are based on specifying an initial number of groups, and iteratively reallocating objects among groups to convergence. In contrast, hierarchical algorithms combine or divide existing groups, creating a hierarchical structure that reflects the order in which groups are merged or divided. In an agglomerative method, which builds the hierarchy by merging, the objects initially belong to a list of singleton sets S1 ,..., S2 , Sn . Then a cost function is used to find the pair of sets {Si , Sj } from the list that is the "cheapest" to merge. [5]

Once merged, Si and Sj are removed from the list of sets and replaced with Si⋃ Sj.This process iterates until all objects are in a single group. Different variants of agglomerative hierarchical clustering algorithms may use different cost functions. Complete linkage, average linkage, and single linkage methods use maximum, average, and minimum distances between the members of two clusters, respectively. Following is the pseudo code of the hierarchical clustering algorithm to explain how it works: [5]

- Compute the proximity matrix containing the distance between each pair of patterns. Treat each pattern as a cluster.
- Find the most similar pair of clusters using the proximity matrix. Merge these two clusters into one cluster. Update the proximity matrix to reflect this merge operation.
- If all patterns are in one cluster, stop. Otherwise go to step 2.

The advantages of the hierarchical clustering algorithms are the reason this algorithm was chosen for discussion. [5] These advantages include:

- Embedded flexibility regarding a level of granularity.
- Ease of handling of any forms of similarity or distance.
- Consequently applicability to any attributes types.
- Hierarchical clustering algorithms are more versatile.

The hierarchical clustering procedures can be divided into two different approaches: agglomerative and divisive.

Agglomerative clustering

The agglomerative approach to cluster analysis, used by the nearest and farthest neighbour algorithms, is a bottom-up clumping approach where we begin with *n* singleton clusters and successively merge clusters to produce the other ones.

Nearest-Neighbour

The following is the nearest-neighbour clustering algorithm:

```
begin
  initialize c; c' = n; $D_i = \{x_i\}$; i = 1,...,n
  do
    c' = c' - 1
    Find nearest clusters $D_i$ and $D_j$
    Merge $D_i$ and $D_j$
  until c = c'
  return c clusters
end
```

To find the nearest clusters in step 5, the following clustering criterion function is used:

$$d_{min}(D_i, D_j) = \min||x - x'||\text{, where } x \in D_i \text{ and } x' \in D_j$$

The merging of the two clusters in step 6 simply corresponds to adding an edge between the nearest pair of nodes in D_i and D_j.

Also, if instead of terminating after a predetermined number of clusters have been obtained; it is possible to set the termination criteria to stop when the distance between nearest clusters exceeds a predetermined threshold, thus becoming the *single-linkage algorithm*.

Some interesting properties of this algorithm is that it will always produce a tree since edges linking clusters always go between distinct clusters, thus no cycles are produced. Also, if the algorithm is allowed to run until all clusters are connected then a spanning tree is generated, or more precisely, a minimal spanning tree since the edges we are inserting between clusters are always the shortest ones in distance.

As a result, the nearest-neighbour algorithm can effectively detect elongated clusters but is not as effective when clusters are compact and of equal size. Therefore we look to the farthest-neighbour algorithm to detect such clusters.

Farthest-Neighbour

The following is the farthest-neighbour clustering algorithm:

```
1.  begin
2.    initialize c; c' = n; $D_i = \{x_i\}$; i = 1,...,n
3.    do
4.      c' = c' - 1
5.      Find nearest clusters $D_i$ and $D_j$
6.      Merge $D_i$ and $D_j$
7.    until c = c'
8.    return c clusters
9.  End
```

To find the nearest clusters in step 5, the following clustering criterion function is used:

$$d_{max}(D_i, D_j) = \max||x - x'||\text{, where } x \in D_i \text{ and } x' \in D_j$$

Also, if instead of terminating after a predetermined number of clusters have been obtained; it is possible to set the termination criteria to stop when the distance between nearest clusters exceeds a predetermined threshold, thus becoming the *complete-linkage algorithm*.

The farthest- neighbour algorithm prevents the grouping of elongated clusters; instead, clusters are composed of complete sub graphs. As a result, the distance of two clusters is determined not by the distance between the two closest nodes (as in the nearest-neighbour algorithm), but by the distance between the two most distant nodes. Thus, when merging the two clusters, edges are added between every pair of nodes in the affected clusters.[4]

This algorithm thus detects clusters which are compact and of roughly equal size, but has difficulties when the clusters are elongated and can generate groupings which are meaningless. [4]

Possible compromises between the two agglomerative approaches suggested above are to use the following criterion to find the nearest clusters:[4]

$$d_{avg}(D_i, D_j) = \frac{1}{n_i n_j} \sum x \in D_i \sum x' \in D_i ||x - x'||$$

$$d_{mean}(D_i, D_j) = ||m_i - m_j||$$

Divisive Clustering

The minimal spanning tree, on the other hand, uses the **divisive approach** which is a top-down approach where we start with one cluster and successively split clusters to produce others. [4]

Minimal Spanning Tree

The minimal spanning tree algorithm is a divisive approach because we are given the minimal spanning tree of all the nodes thus providing us with only one cluster to work with initially. [4]

There exists numerous ways to divide the clusters successively. One way is to simply remove the longest edge and create 2 clusters and then recursively removing the longest edge until the desired number of clusters is obtained.

Another method in selecting which edge to remove is to compare the length of the edges to the lengths of the edges incident upon its nodes. As a result, we can consider an edge to be inconsistent if its length *l* is significantly larger than *l'*, where *l'* is the average length of all other edges incident on its nodes. This method has the advantage of determining clusters which have different distances separating one another.

Another use of the minimal spanning tree is to find dense clusters embedded in a sparse set of points. All that has to be done is to remove all edges longer than some predetermined length in order to extract clusters which are closer than the specified length to each other. If the length is chosen accordingly, the dense clusters are extracted from a sparse set of points easily.

Applications

Cluster analysis has a vital role in numerous fields ranging from biology to machine learning. Its application depends on whether clustering is used as a stepping stool and a basis for future analysis or as a tool for understanding.

Clustering algorithms can be applied in many fields, for instance:

- ***Marketing*****:** finding groups of customers with similar behavior given a large database of customer data containing their properties and past buying records;
- ***Biology*****:** classification of plants and animals given their features;
- ***Libraries***: book ordering;
- ***Insurance*****:** identifying groups of motor insurance policy holders with a high average claim cost; identifying frauds;
- ***City-planning*****:** identifying groups of houses according to their house type, value and geographical location;
- ***Earthquake studies*****:** clustering observed earthquake epicenters to identify dangerous zones;
- ***WWW*****:** document classification; clustering weblog data to discover groups of similar access patterns.

Requirements

The main requirements that a clustering algorithm should satisfy are:

- scalability;
- dealing with different types of attributes;
- discovering clusters with arbitrary shape;
- minimal requirements for domain knowledge to determine input parameters;
- ability to deal with noise and outliers;
- insensitivity to order of input records;
- high dimensionality;
- Interpretability and usability.

Conclusion

In this paper, we have done a comparative study on Hierarchical clustering algorithm. After the comparative study on various papers the following conclusions are obtained.

- Hierarchical clustering algorithm shows good results when using small dataset.[5]
- Running the clustering algorithms using any software gives almost the same results even when changing any of the factors because most software use the same procedures and ideas in any algorithm implemented by them.

References

[1] Michel J.A. Berry Gordon S. Linoff, DM Techniques, Second Edition.

[2] Han J.and Kamber M., Datamining: Concepts and Techniques, Morgan Kaufmann publishers, 2001.

[3] Dr. N. Rajalingam and K.Ranjini, "Hierarchical Clustering Algorithm – A comparative study", International Journal of Computer Applications April 2011.

[4] http://cgm.cs.mcgill.ca/~soss/cs644/projects/siourbas/sect5.html

[5] Osama Abu Abbas, Computer Science department, Yarmouk University, Jordan, "Comparisons between data clustering algorithms", International Arab Journal of Information Technology, Vol.5, No. 3, July 2008