A Comparative Study : BIRCH And CLIQUE

Nagesh Shetty
Department of Information Technology, Shree Rayeshwar Institute of Engineering and Information Technology, Ponda- Goa

Prof. Rudresh Shirwaikar
Department of Information Technology, Shree Rayeshwar Institute of Engineering and Information Technology, Ponda- Goa

Abstract

Clustering is the process of grouping similar objects from a set of different, dissimilar objects to form meaningful sub-classes, called as clusters. Clustering in data mining is a useful technique for discovering patterns and other useful information. It has many applications in the fields of pattern recognition, image analysis and so on. Depending upon the application, clustering can be applied to regular data sets and high dimensional data sets. The most suitable clustering method for analysis of a regular data set is the hierarchical method. BIRCH is an algorithm under this method. Hierarchical clustering is performed by taking Iris data set as an example. Algorithms under hierarchical methods often encounter challenges when faced with high dimensional data set. So, subspace cluster analysis is done for a high dimensional data set. CLIQUE is an algorithm under this method. This paper presents an overview of BIRCH, Clique and sub-space clustering.

Keywords: BIRCH, CF Tree, CLIQUE, subspace clustering, Iris data set, Apriori algorithm.

1. Introduction

Data mining is the extraction of useful knowledge and interesting patterns from a large amount of available information. In this paper, data clustering is examined. Data clustering is the process of grouping objects or data into a cluster. Therefore, a cluster contains a group of similar objects or data. There are five methods of cluster analysis, namely, partitioning, hierarchical, density based, grid based, model based. Clustering can be applied to regular data sets or high dimensional data sets. Regular data sets consist of fewer attributes across one or two dimensions whereas high dimensional data sets contain data having attributes of much higher order spread across N-dimensions.

One of the early methods for analysis of regular data set is hierarchical methods. There are three algorithms under this method. BIRCH, which begins by partitioning objects hierarchically using tree structure; ROCK, merges clusters based on their interconnectivity; and Chameleon, explores dynamic modeling in hierarchical clustering. Among these algorithms, this paper focuses on BIRCH.

Algorithms from hierarchical method if applied to high dimensional data set face challenges and hence, subspace clustering analysis needs to be done. CLIQUE is the first algorithm developed under this method.

2. BIRCH

BIRCH (Balanced iterative Reducing and Clustering Hierarchies) is an unsupervised data mining algorithm which uses the agglomerative approach for clustering large amount of numerical data. Agglomerative hierarchical clustering is a bottom up clustering method where clusters have sub-clusters which in turn have sub-clusters.

BIRCH introduces two concepts, clustering feature and clustering feature tree (CF Tree), which are used to summarize cluster representations. This helps the algorithm achieve a good speed and scalability and also makes it effective in large data sets and in incremental and dynamic clustering of incoming objects.

A Clustering Feature is a triple vector, summarizing the information about clusters of objects or data. It consists of ‘n’ as number of points in a cluster, LS as linear sum of n points and SS as sum square of n points. Given n d-dimensional points in a cluster then CF of cluster is given as

$$CF = \{n, LS, SS\}$$

Clustering Features are additive. For example consider two clusters C1 and C2 with clustering feature as CF1 and CF2 respectively. The clustering feature for the cluster C formed by merging C1 and C2 can be simply given as CF1+CF2. These are useful for making clustering decisions in BIRCH.

A CF Tree is a height balanced tree that stores the clustering features for hierarchical clustering as shown in figure 1. The internal or non-leaf nodes store sum of CFs of their children. A CF tree has two parameters: branching factor B and threshold T.

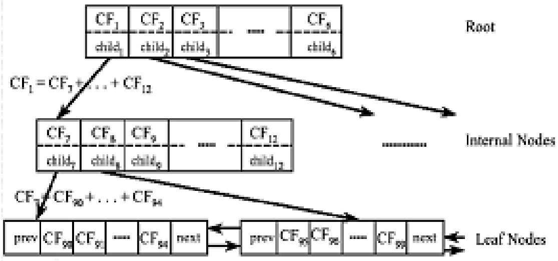

CF1 CF2 CF3 ... CF6
child1 child2 child3 child6
Root
CF1 = CF7 + . . . + CF12
CF7 CF8 CF9 ... CF12
child7 child8 child9 child12
Internal Nodes
CF7 = CF90 + . . . + CF94
prev CF90 CF91 ... CF94 next
prev CF95 CF96 ... CF99 next
Leaf Nodes

Figure 1: CF Tree Structure.

The branching factor specifies the maximum number of children per non-leaf node while threshold specifies the maximum diameter of sub clusters at leaf node. Both of these influence the size of the tree.

The algorithm operates in following four steps as:

1. Scans all data and builds an in memory CF Tree using the given amount of memory and available space on the disk.
2. Condense the original tree in a desirable range by building a smaller CF Tree.
3. Use a global or semi-global clustering algorithm to cluster all leaf entries.
4. This is optional phase wherein additional passes can be made over the data set to refine the clusters formed.

In the first phase, the CF tree is built dynamically as objects are inserted. The CF tries to reflect the clustering information of the data set as finely as possible under the memory limit where crowded points are grouped as fine as sub clusters, and sparse data points are separated as outliers. Phase 2 is an optional phase. Phase 2 also plays the role of cushion between Phase 1 and Phase 3. It is observed that global or semi-global clustering methods that we adopt in phase 3 will perform well in terms of speed and quality. In Phase 3 we obtain set of clusters that captures the major distribution patterns in the data. Phase 4 is optional and incurs the cost of additional passes over the data to correct those inaccuracies and refine the cluster further. Phase 4 also provides the option of discarding outliers.

BIRCH tries to produce the best cluster within the available resources. Given a limited amount of main memory, an important consideration is to have a reduced time for I/O. A single scan of the data set yields a good cluster and one or more scans will improve the quality of cluster further. The complexity of BIRCH is O(n) where n is the number of objects to be clustered.

The weakness of this algorithm:

- Since each node in a CF-Tree can hold only a limited number of entries due to the size, a CF-Tree node does not always correspond to what a user may consider a natural cluster.
- The algorithm does not perform well if the clusters are not spherical in shape because it uses the notion of radius or diameter to control the boundary of the cluster.

The strengths of this algorithm:

- BIRCH incrementally and dynamically incoming, multidimensional data points to try to produce the best quality clustering with the available time and resources.
- The algorithm can typically find a good clustering with a single scan of the data and improve the quality further with a few additional scans.
- BIRCH is the first algorithm to be proposed in the database area that handles noise effectively.

3. SUBSPACE CLUSTERING

Subspace clustering is done with respect to high dimensional data. Some clustering algorithms fail when they encounter high dimensional data. This is due to the fact that when dimensionality increases, only a small number of dimensions are relevant for certain clusters, but data in irrelevant dimensions may act as noise and mask the formation of the real clusters. Also, when dimensionality increases data points become sparse and all points located at different dimensions can be considered as equally placed, and the vital factor needed for cluster analysis i.e. distance measure, becomes meaningless.

Subspace clustering is related to attribute subset selection which has shown its strength at high dimensional clustering. From the experiments, it has been observed that different subspaces have different meaningful clusters. It finds out different clusters among different subspaces of same data set. The problem is to find out subspaces effectively and efficiently. To serve this purpose, CLIQUE algorithm has been designed. It was the first algorithm for dimensional-growth subspace clustering in high dimensional space.

4. CLIQUE

CLIQUE (CLustering In QUEst) as mentioned above, is the first algorithm for dimensional-growth subspace clustering in high dimensional space. The process of clustering starts with a single dimension and grows upwards to higher dimensions. Clique partitions the m-dimensional data space into non-overlapping rectangular units. Dense units are identified from these units. A unit is dense if the sum of total data points in a unit exceeds the input parameter. The clusters formed from original data spaces uses Apriori property. The property is stated as following: if a k-dimensional unit is dense, then so are its projections in (k-1) dimensional space.

The clusters are formed based on the following algorithm.

1. The original data space is divided it into sub spaces
2. Partitions the data space into non overlapping rectangular units
3. A unit is dense if the sum of total data points in a unit is greater than some threshold value provided initially by the user.
4. Thus, identifies dense units from the grid structure.
5. From the rectangular connected regions group together the maximum set of dense region to identify the cluster.
6. Subsequently generate minimal descriptions for the clusters.

CLIQUE is effective as it finds out the subspaces of the highest dimensionality such that high quality clusters exist only in those subspaces. Obtaining meaningful clustering results is dependent on proper tuning of grid size and the threshold density. This is particularly difficult because the grid size and density threshold used across the combinations of dimensional in the data set. Instead of using a grid of a set of fixed bins for each dimension, we can use an adaptive, data-driven strategy to dynamically determine bins for each dimension based on data distribution statistics.

The weakness of this algorithm:

- Due to the user defined threshold value useful part of cluster may be missed.
- The algorithm is insensitive to order of input objects.

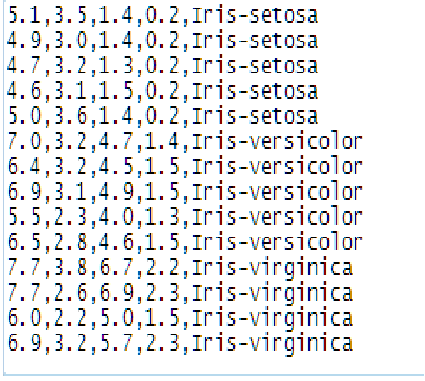
The strengths of this algorithm:

- Finds out the dense region automatically.
- The algorithm is linearly scalable with size of input data provided.
- Provides good scalability even if the number of dimension increases.

5. Experimental work

A. Data sets used

- **Iris data set:** The iris flower data set is a multivariate data set. This data set is also called as Anderson's Iris data set as he collected to quantify the morphologic variation of Iris flower of three related species. This data set can be used for cluster analysis. The commonly clusters formed are of Iris setosa, Iris virginica, Iris versicolor. This makes the data set a good example to explain the difference between supervised and unsupervised techniques in data mining.



```
5.1,3.5,1.4,0.2,Iris-setosa
4.9,3.0,1.4,0.2,Iris-setosa
4.7,3.2,1.3,0.2,Iris-setosa
4.6,3.1,1.5,0.2,Iris-setosa
5.0,3.6,1.4,0.2,Iris-setosa
7.0,3.2,4.7,1.4,Iris-versicolor
6.4,3.2,4.5,1.5,Iris-versicolor
6.9,3.1,4.9,1.5,Iris-versicolor
5.5,2.3,4.0,1.3,Iris-versicolor
6.5,2.8,4.6,1.5,Iris-versicolor
7.7,3.8,6.7,2.2,Iris-virginica
7.7,2.6,6.9,2.3,Iris-virginica
6.0,2.2,5.0,1.5,Iris-virginica
6.9,3.2,5.7,2.3,Iris-virginica
```

Figure 2: Snapshot of Iris dataset.

As shown in the above figure, the first column of the dataset indicates the sepal length, second column indicates the sepal width, third column indicates the petal length, fourth column indicates the petal width and the last column indicates the names of three Iris flower species.

This is a conventional dataset suitable for the BIRCH algorithm.

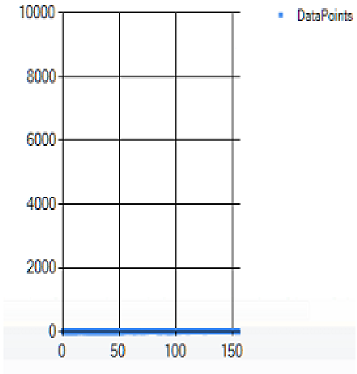
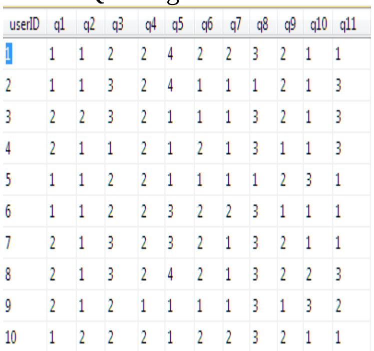


Figure 3: Result of BIRCH on Iris data set.

The above graph indicates the output of BIRCH algorithm on Iris data set where in the blue points indicate the Iris data. Since the output of BIRCH is a single cluster, which is shown by a straight line which indicates the Iris data

- Mobile Phone Survey

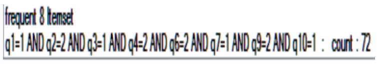
A survey was conducted among 170 students of our Institute wherein eleven questions were asked related to mobile phone usage. This data was then entered into SQL database. The database table consisted of twelve columns and one hundred and seventy one rows as shown in the figure 4. CLIQUE algorithm was applied on this database. This is a high dimensional dataset suitable for the CLIQUE algorithm.



userID	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11
1	1	1	2	2	4	2	2	3	2	1	1
2	1	1	3	2	4	1	1	1	2	1	3
3	2	2	3	2	1	1	1	3	2	1	3
4	2	1	1	2	1	2	1	3	1	1	3
5	1	1	2	2	1	1	1	1	2	3	1
6	1	1	2	2	3	2	2	3	1	1	1
7	2	1	3	2	3	2	1	3	2	1	1
8	2	1	3	2	4	2	1	3	2	2	3
9	2	1	2	1	1	1	1	3	1	3	2
10	1	2	2	2	1	2	2	3	2	1	1

Figure 4: Collected data from students.

The above figure represents a brief information about the data set collected from the survey. The first column indicates the ID assigned to every person who participated in the survey. The following eleven columns indicate the question numbers from one to eleven. Depending on the options selected by each person for each question, numbers from one to four were assigned. For example, the second row in the above table shows that user1 selected option a for question one, option b for question 3, option d for question 5 and so on.



```
frequent 8 Itemset
q1=1 AND q2=2 AND q3=1 AND q4=2 AND q6=2 AND q7=1 AND q9=2 AND q10=1  :  count : 72
```

Figure 4: Output of CLIQUE on the mobile phone usage.

The above figure it can be seen that out of one hundred and seventy students, seventy two are males, who lie in the age group between 15-20, frequently use their phones for voice calls, they spend less than thirty minutes on voice calls, their average montly bill is between Rs. 101 to Rs 500 and they also keep their phone on silet mode when at college or at work.

6. Conclusion

Regular dataset can be efficiently clustered using the algorithm BIRCH. It efficiently clusters even datasets with a large number of entries and is thus, linearly scalable. The algorithm produces good clusters within a single pass over the entire dataset. It even handles noise or outliers effectively. However the algorithm faces challenges when encountered with high dimensional data. In this case, the principle of clustering based on distance measurements is rendered meaningless.

CLIQUE the algorithm CLIQUE does not take into account distance measurements between points but is density based. It also identifies sub spaces from the N- dimensional data space automatically and efficiently. Hence it is suitable for clustering of high dimensional datasets.

Thus, BIRCH and CLIQUE are suitable algorithms in their respective domains.

Table 1: Differences between BIRCH and CLIQUE.

	BIRCH	CLIQUE
Type of algorithm	Hierarchical	Grid based
Uses	CF Tree	Apriori principle
Clustering method	Conventional	Sub space
Data set type	Regular mutivalued	Multi attribute

Scalability	Highly scalable	Highly scalable
Noise	Handles noise effectively	Handles noise effectively

7. References

[1] Jiawei Han, Micheline Kamber, Data Mining Concepts and Techniques, 2nd ed, Morgan Kauffman.

[2] Pang Ning Tan, Michael Steinbach, Vipin Kumar, Introduction to Data Mining

[3] T. Zhang, Raghu Ramakrishnan and Miron Livny, “BIRCH: A new clustering algorithm and its applications”.

[4] Glenn Fung, “A comprehensive overview of clustering algorithms”.

[5] Khaled Alsabti, sanjay ranka and Cineet Singh “An efficient K-mean clustering algorithm..

[6] Manam Rehman, Syed Atif Mehdi, “Comparison of density based clustering algorithms”.

[7] Raghunath Kar & Susant Kumar Dash., “A study on high dimensional clustering by using clique”.

[8] Hans Peter Kriegel, Peer Kroger and Arthur Zimek, “Clustering High-Dimensional Data: A Survey on subspace Clustering, Pattern-Based Clustering, and Correlation Clustering.

[9] Rakesh Agrawal, Johannes Gehrke, Dimitrios Gunopulos and Prabhakar Raghavan, “Automatic Subspace Clustering of High Dimensional Data for Data Mining Applications”.

[10] Hans-Peter Kriegel, Peer Kroger, Matthias Renz and Sebastian Wurst, “A Generic Framework for Efficient Subspace Clustering of High-Dimensional Data”.

[11] Lance Parsons, Ehtesham Haque and Huan Liu, “Evaluating Subspace Clustering Algorithms”.

[12] Michael Steinbach, Levent Ertöz and Vipin Kumar, “The Challenges of Clustering High Dimensional Data”.

[13] D.Pramodh Krishna, A.Senguttuvan and T.Swarna Latha “Clustering on Large Numeric Data Sets Using Hierarchical Approach: BIRCH”.

[14] Aminur Rashid, Irvanizam Zamanhuri and Philipp Volgger, “BIRCH Clustering Algorithm”.

[15] Hemlata Sahu, Shalini Shrma and Seema Gondhalakar, “A Brief Overview on Data Mining Survey”.