EFFICIENT ALGORITHM FOR MINING FREQUENT ITEMSETS USING CLUSTERING TECHNIQUES

D.Kerana Hanirex
Research Scholar
Bharath University

Dr.M.A.Dorai Rangaswamy
Professor ,Dept of IT,
Easwari Engg.College

Abstract

Now a days, Association rule plays an important role. The purchasing of one product when another product is purchased represents an association rule. The Apriori algorithm is the basic algorithm for mining association rules. This paper presents an efficient Partition Algorithm for Mining Frequent Itemsets(PAFI) using clustering. This algorithm finds the frequent itemsets by partitioning the database transactions into clusters. Clusters are formed based on the similarity measures between the transactions. Then it finds the frequent itemsets with the transactions in the clusters directly using improved Apriori algorithm which further reduces the number of scans in the database and hence improve the efficiency.

Keywords:Association rule,Apriori algorithm,frequent Itemset ,clustering

1.INTRODUCTION

Mining association rule is one of the recent data mining research. Association rules are used to show the relationships between data items. Association rules are frequently used in marketing, advertising and inventory control .Association rules detect common usage of items. This problem is motivated by applications known as market basket analysis to find relationships between items purchased by customers [4], that is, what kinds of products tend to be purchased together.This paper presents an efficient Partition Algorithm for Mining Frequent Itemsets (PAFI) using clustering technique. This algorithm finds the frequent itemsets by partitioning the database transactions into clusters. Clusters are formed based on the similarity measures between the transactions. Then it finds the frequent itemsets with the transactions in the clusters directly using the improved Apriori algorithm which further reduces the number of scans in the database and hence improve the efficiency.

2.ASSOCIATION RULE PROBLEM A database in which an association rule is to be found is viewed as set of tuples, where each tuple contain a set of items. Each item represents an item purchased while each tuple is the list of items purchased at one time. The support(s) of an item is the percentage of transactions in which that item occurs. Given a set of items $I=\{I_1,I_2,\ldots I_m\}$ and a database transactions $D=\{t_1,t_2\ldots\ldots t_n\}$ where $t_i=\{I_{i1},I_{i2}\ldots I_{ik}\}$ and $Iij \in I$, an association rule is an implication of the form $X \Rightarrow Y$ where $X, Y \subset I$ are sets of items called itemsets and $X \cap Y = \Phi$ .The confidence or strength (α) for an association rule $X \Rightarrow Y$ is the ratio of the number of transactions that contain $X \cup Y$ to the number of transactions that contain X.

The association rule problem is to identify all association rules with a minimum support and confidence. The efficiency of an association rule algorithms usually discussed with respect to the number of scans of the database that are required and the maximum number of itemsets that must be counted.

The most common approach to find association rules is to break up the problem into 2 parts

1.Find Large Itemsets

2.Generate rule from the frequent Itemsets

A Large (Frequent) Itemset is an Itemset whose number of occurance is above the threshold (s).

3.APRIORI ALGORITHM

The Apriori Algorithm is the most well known association rule algorithm and it is used in most commercial products. It uses largest itemset property[1].

"Any subset of a large itemset must be large"

The basic idea of Apriori algorithm is to generate item sets of a particular size and then scan the database to count these to see if they are large. Only those candidates that are large are used to generate candidates for the next scan. Li is used to generate next Ci+1. L represent Large

Itemset. C represents candidate items. All singleton itemsets are used as candidates in the first pass. The set of large item sets of the previous pass,Li-1 is joined with itself to determine the candidates. Individual itemsets must have all but one item in common in order to be combined.

4.CLUSTERING AND PARTITIONING

Clustering is a data mining technique used to place data elements into related groups without advance knowledge of the group definitions. Clustering can be considered the most important unsupervised learning problem; so, as every other problem of this kind, it deals with finding a structure in a collection of unlabeled data. A cluster is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters. Clustering algorithms may be classified as listed below:

- Exclusive Clustering
- Overlapping Clustering
- Hierarchical Clustering
- Probabilistic Clustering

Data set partitioning algorithm is the basis of the various parallel association rule mining algorithm and distributed association rule mining algorithm. The partition algorithm [5]-[6]-[7] is based in the observation that the frequent sets are normally very few in number compared to the set of all itemsets. In recent years several fast algorithms including Apriori [7] and Partition [6] for generating frequent itemsets have been suggested in the literature [9]-[10]-[11]-[12]-[13]. A critical analysis of these has led the authors to identify the following limitations/shortcomings in them. In [8]

By taking advantage of the large itemset property, this is that a large itemset must be large in at least one of the partitions. This idea can help to design algorithms more efficiently than those based on looking at the entire database.

Partitioning algorithms may be able to adapt better to limited main memory. Each partition can be created such that it fits in to main memory. In addition it would be expected that the number of itemsets to be counted per partition would be smaller than those needed for the entire database.

By using partitioning, cluster based and/or distributed algorithms can be easily created, where each partitioning could be handled by a separate machine.

Incremental generation of association rules may be easier to perform by treating the current state of the database as one partition and treating the new entries as a second partition.

As the result, if the set of transactions are partitioned in to smaller segments such that each segment can be accommodated in the main memory, then the set of frequent sets of each of these partitions can be computed. Therefore this way of finding the frequent sets by partitioning the database may improve the performance of finding large itemsets in several ways.

Various approaches for generating large item sets have been proposed based on partitioning the set of transactions. The clusters are formed by partitioning the set of transactions based on the similarity measures between the transactions. Transactions are iteratively merged in to the cluster that are closest.

This PAFI algorithm suggests the number of clusters (NOC)that are formed based on the number of transactions or total count of transactions (COT). By assumption, it can be calculated as the ratio of number of transactions to some random natural number N. The transaction having largest number of items will be put in the first cluster CL_1 .The transaction having the next highest number of items will be put in the next cluster CL_2. This process is repeated until each cluster have at most one itemset. Next all the transactions in the database are scanned and put the transaction into the cluster that have the highest similarity measures with the existing itemset. The similarity is measured based on that the number of items that are in common with the existing itemset. Then the number of transactions with in each cluster is counted.

In order to find the Largest item set it is enough to go through the transactions with in the clusters. The cluster that have the total number of transactions less than some threshold value will be deleted. For finding the large itemsets it is enough to go through the transactions with in the clusters. There is no need to go through the entire database again. Hence it reduces the redundant database scan and improves the efficiency. If we apply the

Improved APRIORI algorithm to find large itemsets after partitioning it will further reduces the number of scans and improves efficiency.

5. ALGORITHM

5.1. PAFI (Partition Algorithm for Frequent Itemsets) ALGORITHM

Begin
Number of clusters(NOC)=count of transactions(COT)/N //N is random natural number
FOR i= 1 to NOC DO BEGIN
FOR each cluster C_i DO BEGIN
FOR each transaction $t \in D$ DO BEGIN
Find t such that t having highest number of items
Put t in C_i
END
END
Return Clusters with 1 itemset.

5.2.Improved Apriori Algorithm

Improved Apriori algorithm mines frequent itemsets with out new candidate generation[2].In this algotithm we are computing the frequency of frequent k-itemsets from k-1 itemsets. If k is greater than the size of the transaction T,there is no need to scan the transaction T which is generated by (k-1) itemsets according to the nature of Apriori algorithm, and we can remove it.

Improved Apriori Algorithm

//AprioriGen to find frequent itemset

FOR i= 1 to NOC DO BEGIN
FOR each cluster C_i DO BEGIN
If count(t) > threshold then
FOR each transaction $t \in C_i$ DO BEGIN
FOR each item p in t DO BEGIN
// for singleton itemsets
Find count(p)
END
FOR each item p,q in t DO BEGIN
// for 2 itemsets
Find count(p,q)
END
END
If count(p) <minsupport then delete t from C_i;
Repeat Until frequent itemsets occurs
Else
Skip C_i
END
$L=LUL_{i;}$
END
END
Return L; //L gives set of all frequent itemsets

This improved algorithm efficiently finds the frequent itemsets .Thus reduces the number of scans and save space also the computing time is improved.

6. EXPERIMENTAL RESULTS

This is an example based on the following transactions in the database D. First we are applying Partition algorithm(PAFI) to find clusters then we are applying Improved Apriori algorithm to find the frequent itemsets.

Steps:

1.For a given set of transactions in the database D, it applies partition algorithm in order to find clusters based on the number of transactions. Here we are getting 2 clusters CL_1 and CL_2.

2.Here CL_2 has less number of transactions that is less than the threshold value so we are deleting the transactions in CL_2 and we are concentrating on the transactions in CL_1.

3.Now apply the improved Apriori algorithm ,by finding the count of an each item from D_1.Since scanning is not done using the entire database transactions D it improves the efficiency. These set of items will be considered as candidates C_1.These transactions will be considered as D_1.

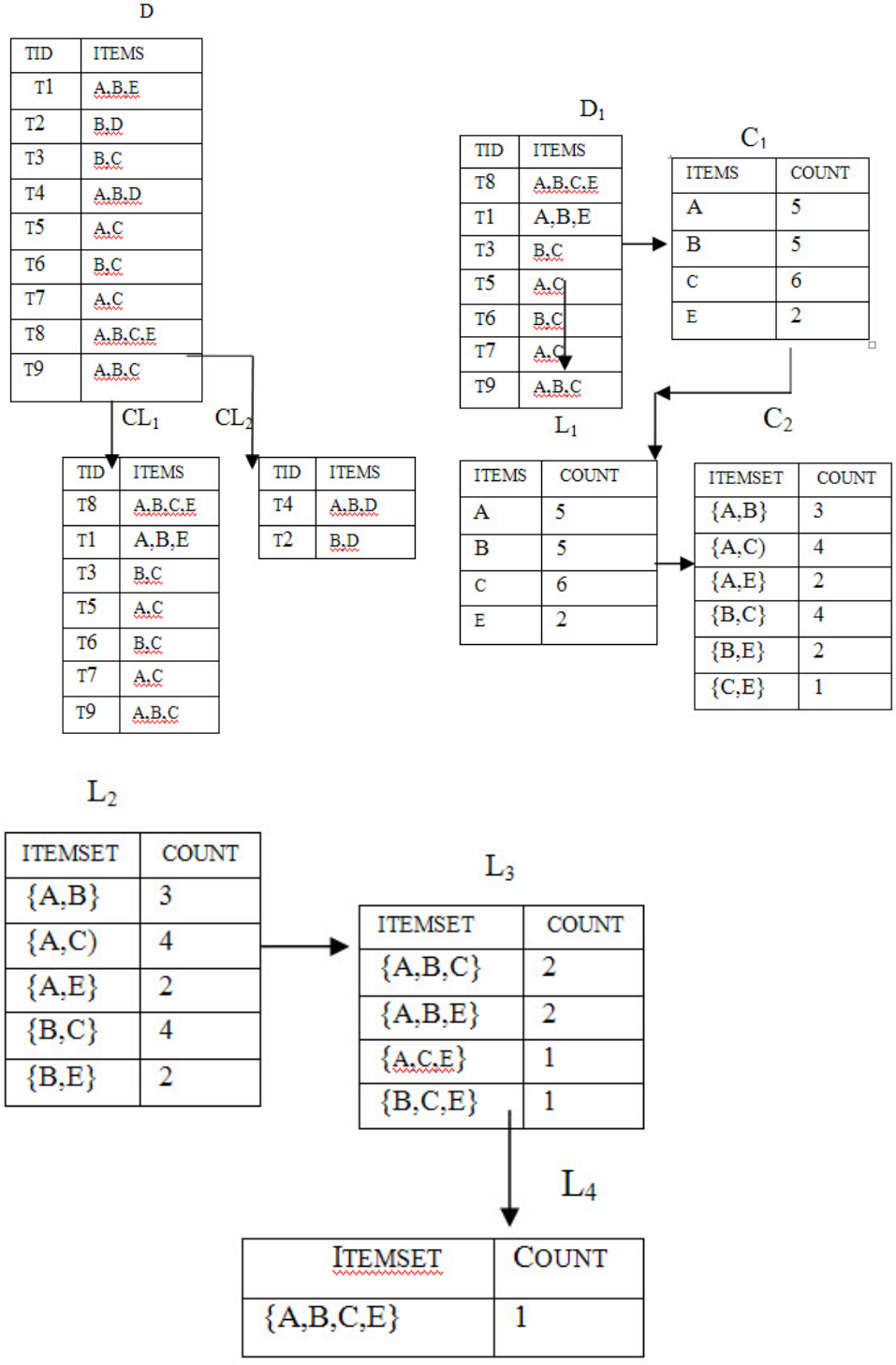

Fig1: Generation of Frequent Itemsets

4.D_1 is scanned to generate C_2 candidates and find the count of each candidate.

5.Compare candidate support with minimum support.The candidates having less count than the minimum support will be deleted..The above process is repeated for C_3.

6.This will be generated for C_k until C_{k+1} becomes empty.

7.CONCLUSIONS

In this paper, the Partition Algorithm for Frequent Itemset (PAFI) is proposed before applying Improved Apriori Algorithm. This algorithm reduces the number of scans in the database and improves efficiency and computing time by taking the advantage of clustering technique. By experiment results, it can obtain higher efficiency.

8.REFERENCES

[1] Lee-Wen Huang ,Ye-In Chang,"A Graph-Based Approach for Mining Closed Large Itemsets" National Sun Yat-Sen University.

[2] Sheng Chai, Jia Yang and Yang Cheng, "The Research of Improved Apriori Algorithm for Mining Association Rules" In Proceedings of the Service Systems and Service Management, 2007 International Conference, 9-11 June 2007 pages : 1 – 4.

[3] Ja-HwungSu,Wen-Yang Lin "CBW: An Efficient Algorithm for Frequent Itemset Mining" In Proceedings of the 37th Hawaii International Conference on System Sciences – 2004.

[4] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," in Proceedings of the 20th VLDB Conference, 1994, pp. 487−499.

[5] Arun K Pujari. Data Mining Techniques (Edition 5):Hyderabad, India: Universities Press (India) Private Limited, 2003.

[6] Margatet H. Dunham. Data Mining, Introductory and Advanced Topics: Upper Saddle River, New Jersey: Pearson Education Inc., 2003.

[7] Jiawei Han. Data Mining, concepts and Techniques: San Francisco, CA: Morgan Kaufmann Publishers.,2004.

[8] Akhilesh Tiwari, Rajendra K. Gupta, and Dev Prakash Agrawal "Cluster Based Partition Approach for Mining Frequent Itemsets" In Proceedings of the IJCSNS International Journal of computer Science and Network Security, VOL.9 No.6, June 2009

[9] R.K. Gupta. Development of Algorithms for New Association Rule Mining System, Ph.D. Thesis, Submitted to ABV-Indian Institute of information Technology & Management, Gwalior, India, 2004.

[10] M. Houtsma and A. Swami. Set Oriented Mining for Association Rules in Relational Databases. In Proceedings of 11th International conference on Data Engineering, 1995, pp 25-33,.

[11] Agarwal R., Imielinski T., and Swami A. Mining associations between sets of items in massive databases. In Proceedings of the ACM SIGMOD International Conference on Management of Data, Washington D.C. , May 1993, pp. 207-216.

[12] M. Houtsma and A. Swami, Set Oriented Mining for Association Rules in RelationalDatabases. In Proceedings of 11th IEEE International Conference on Data Engineering, 1995, pp : 25-33.

[13] Rakesh Agrawal and R . Srikant . Fast Algorithm for Mining Association Rules in Large Databases, Proceedings of the 20th International Conference on Very Large Databases, Santigo, Chile, 1994, pp 487-499.