International Journal of Advanced Computer Technology (IJACT)
ISSN:2319-7900

PERFORMANCE ORIENDED PARTITION ALGORITHM FOR MINING FREQUENT ITEMSET IN TWO DIMENSIONAL APPROACHES

Dr.S.Sivasubramanian[1] Dr. U.Janardhana Raju[2]
[1]*Professor, Department of Information Technology*
Dhanalakshmi College of Enginnering, Chennai-73, Tamil Nadu, India.
[2] Testing Team Lead @ ValueSource Technologies Pvt Ltd
Chennai-73, Tamil Nadu, India.

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) and Two Dimensional Approach for Mining Frequent Itemsets(TDFI).This algorithm finds the frequent itemsets by dividing the database transactions into various partitions. Then for each partition it finds the frequent itemsets using a two dimensional approach which further reduces the number of scans in the database and hence improve the efficiency.

Keywords :Association rule,Apriori algorithm, frequent Item set , partitioning

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 the 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) and Two Dimensional Approach Algorithm for Mining Frequent Itemsets (TDFI).This algorithm finds the frequent itemsets by dividing the database transactions into various partitions. Then for each partition it finds the frequent itemsets using a two dimensional approach 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 a 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 $I=\{I_1,I_2,\ldots.In\}$ and a database transactions $D=\{t_1,t_2\ldots\ldots t_m\}$ where $t_i=\{I_{i1},I_{i2}\ldots.I_{in}\}$ 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 re-

spect 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 Item sets

A Large (Frequent) Itemset is an Itemset whose number of occurrence 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 item set 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. L_i is used to generate next C_{i+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,L_{i-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. PARTITIONING

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 on 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 partitions are formed by dividing the set of transactions based on the similarity measures between the transactions. Transactions are iteratively merged in to the partition that are closest.

This PAFI algorithm suggests the number of transactions(λ) in each partitions by taking the number partitions (P) as user specified input. By assumption, it can be calculated as the ratio of total number of transactions to some random natural number P<m. The first λ transactions will be put in the first partition P_1. The next λ transactions will be put in the next partition P_2 and so on.

In order to find the Largest item set it is enough to go through the transactions with in the partitions. There is no need to go through the entire database D again. Hence it reduces the redundant database scan and improves the efficiency. If we apply Two Dimensional Approach Algorithm for Mining Frequent Item sets(TDFI) to find large item sets after partitioning it will further reduces the number of scans and increases the efficiency.

5. ALGORITHM

5.1. PAFI (Partition Algorithm for Frequent Itemsets) ALGORITHM

The proposed algorithm (PAFI) divides the database into P partitions with λ transactions in each partition.

Algorithm:
Input:
1.Database D
2.Number of partitions P
Output:
Partitions with λ transactions
Begin
Number of transactions in each partition (λ)= Total transactions in D/P //P<m is the random natural number

FOR each partition P_i DO BEGIN
Take λ transactions in P_i
Put each t_i in P_i
END

Return partitions with λ transactions.

5.2. TDFI (Two Dimensional Approach Algorithm for Mining Frequent Itemsets)

The problem of mining frequent itemset is to find all itemsets that are greater than the user specified minimum support count (ie) min_sup. The support count for each partitions are calculated. This algorithm efficiently finds the frequent itemsets .Thus reduces the number of scans and save space also the computing time is improved.

Algorithm:
Input:
1.Database D
2.Minimum support count min_sup

Output:
Frequent Itemsets
Algorithm:

//Algorithm to find frequent itemset

FOR i= 1 to P DO BEGIN
FOR each partition P_i DO BEGIN
FOR each transaction $t \in P_i$ DO BEGIN

Create a new row with the different number of items in D and mark it as 't' if purchased 'f' if not purchased

FOR each item $I_i \in P_i$ DO BEGIN
// for singleton itemsets
Find supcount

END
If supcount <min_sup then delete I_i from P_i;

FOR each item I_i, $I_j \in P_i$ in t DO BEGIN
// for 2 itemsets
Perform AND operations
Find supcount
END
If supcount <min_sup then delete I_iI_j from P_i;
END

Repeat for n itemsets Until frequent itemsets occurs
END

$L=LUL_i$;
END
Return L; //L gives the set of all frequent itemsets

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 partitions then we are applying Two dimensional approach algorithm to find the frequent itemsets(TDFI).

Steps:

1.For a given set of transactions in the database D, it applies PAFI algorithm to find partitions with λ transactions in each partition. Here we are getting 2 partitions P_1 and P_2.

2.For each partition and for each transaction create a new row with the different number of items in D and mark it as 't' if the item is present in that transaction otherwise mark it as 'f' .

3. calculate the support count for all items in L_1.Delete the items having support count less than the minimum support from L_1.

4.Perform AND operations to calculate 2-Itemsets and calculate the support count for all items in L_2.Delete the items having support count less than the minimum support from L_2.

D

TID	ITEMS
T1	A,B,E
T2	B,D
T3	B,C
T4	A,B,D
T5	A,C
T6	B,C
T7	A,C
T8	A,B,C,E
T9	A,B,C

P_1

TID	ITEMS
T1	A,B,E
T2	B,D
T3	B,C
T4	A,B,D
T5	A,C

P_2

TID	ITEMS
T6	B,C
T7	A,C
T8	A,B,C,E
T9	A,B,C

L_1 Large ItemSets-1

	A	B	C	D	E
T1	t	t	f	f	t
T2	f	t	f	t	f
T3	f	t	t	f	f
T4	t	t	f	t	f
T5	t	f	t	f	f
	3	4	2	2	1

Take Min_sup=2

ISSN:2319-7900

L_2 Large ItemSets-2

	AB	AC	AD	BC	BD	CD
t1	t	f	f	f	f	F
t2	f	f	f	f	t	F
t3	f	f	f	t	f	F
t4	t	f	t	f	t	F
t5	f	t	f	f	f	f
	2	1	1	1	2	0

L_3 Large ItemSets-3

	ABD
T1	f
T2	f
T3	f
T4	t
T5	f
	1

The frequent itemsets are AB and BD

5.Repeat the steps for 3-Itemsets and so on until we get the frequent itemsets.L gives the frequent itemset from all partitions.

7.CONCLUSIONS

In this paper, the Partition Algorithm for Mining Frequent Itemset (PAFI) and TDFI (TwoDimensional approach Algorithm for Mining Frequent Itemsets) are proposed to find frequent itemsets. This algorithm reduces the number of scans in the database and improves efficiency and reduces the computing time by taking the advantage of partitioning technique. By experiment results, it can obtain higher efficiency. This happens to be an exponential function in terms of number of items. This happens to be an exponential function in terms of number of items. We verified the complexity of finding frequent itemset by comparing to traditional Apriori Algorithm. Hence this algorithm is an orthogonal version of the standard Apriori algorithm.

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.
[14]Kerana Hanirex.D and M.A.Dorai Rengaswamy,"Efficient Algorithm for Mining Frequent Itemsets using Clustering techniques",Proceedings of the National Conference on,2010 pp

Biographies

Dr S.Sivasubramanain,M.Tech(CSE).,Ph.D (CSE) as an Professor from Department of IT of Dhanalakshmi College Of Engineering, Chennai,Tamil Nadu. He has more than 14 years of teaching and research experience and his areas of specialization are mobile Computing, Database Management System, Computer Networks, Networks Security and Data Mining.

Dr. Janardhana Raju Ph.D 6+ years of experience in IT industry with specialization in Mainframes & TestingRich experience in Testing vertical, with strong expertise in software testing best practices, software test management, test planning, test execution and test status reporting Working as **Testing Team Lead** at **ValueSource Technologies Pvt Ltd**, Chennai, Member of **KBC Group NV**, Belgium, since Oct 2007