Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.727 – 732

RESEARCH ARTICLE

ASSOCIATION RULE MINING IN DISTRIBUTED DATABASE SYSTEM

E.Deenadayalan
II year M.Tech/CSE
Bharath University,
Chennai-73
deenajed@gmail.com

D.Kerana Hanirex
Asst.Professor/CSE,
BharathUniversity,
Chennai-73
keranarobinson@gmail.com

Dr.K.P.Kaliyamurthie
HOD/Dept of CSE,
Bharath University,
Chennai-73
kpkaliyamurthie@gmail.com

Abstract: Data mining is one of the crucial research areas. Among this, discovery of association rules is an important research topic. This paper implements a distributed database algorithm (DD) for mining association rules. The efficiency of this algorithm is compared with the standard FP-Growth algorithm. This algorithm produces the same result as that of FP-Growth with higher efficiency and accuracy. This paper is tested against with the connect data sets and hence prove its efficiency and accuracy.
Index terms: data mining, association rule mining, distributed database

I. INTRODUCTION

With increasing applications on iternset data mining has been applied to the distributed environment of large amount of data. Since data are located in different places ,an efficient algorithm is needed to mine the large amount of data[1].This paper proposes distributed database algorithm (DD) for mining the association rules by finding the local frequent itemset and then generate global frequent itemset. The efficiency of this algorithm is compared with the standard FP-Growth algorithm. This paper is tested against with the connect data sets and hence prove its efficiency and accuracy.

II.RELATED WORK

R.Agarwal proposes mining association rule for large data sets[2].Various fast algorithm for distributed system is proposed. Cheun.D.W [3] proposes a fast distributed algorithm for mining association rules by reducing the number of messages passed. Thabet Slimani proposes current trend in association mining and compare the performance of different algorithms [4].

III. ASSOCIATION MINING

Association mining is a very popular research area and it is used to find the interesting patterns and rules from the large data bases understanding.

- Item**:** It is a field of the transaction database. It is denoted by I_j to denote the item.
- Transaction: It is corresponding to a record of the data base D.It is denoted by the identifier called TID.A set of transaction t_i constitutes a database D D=$\{t_1,t_2\ldots,t_n\}$
- Item set: It is the sets of whole item in a transaction database D where I= $\{ i_1,i_2\ldots,i_m \}$
- Association rule is an implication of the form: $X \rightarrow Y$, where $X, Y \subset I$, *and* $X \cap Y = \varnothing$
- Support: Support is an indication of how frequently the items appear in the database. The rule holds with support sup in T the transaction if sup% of transactions contain $X \cup Y$.

  $\text{Sup} = \Pr(X \cup Y)$.
- Confidence: The rule holds in *T* with confidence *conf* if *conf*% of transactions that contain *X* also contain *Y*.

  $Conf = \Pr(Y \mid X)$

  An association rule is a pattern that states when *X* occurs, *Y* occurs with certain probability.
- Support count: The support count of an item set *X*, denoted by *X.count*, in a data set *T* is the number of transactions in *T* that contain *X*. Assume *T* has *n* transactions.

  Then,

$$support = \frac{(X \cup Y).count}{n}$$

$$confidence = \frac{(X \cup Y).count}{X.count}$$

- Frequent itemset: An itemset, whose support is not lower than the minimum support

IV. FP TREE

FP Tree algorithm is used to find the frequent itemset without the candidate itemset generation.
Two step approach:
Step 1: Build a compact data structure called the FP-tree
Step 2: Extracts frequent itemsets directly from the FP-tree

FP-Tree Construction

FP-Tree is constructed using 2 passes over the data-set:

Pass 1:
(i)Scan data and find support for each item.
(ii)Discard infrequent items.

(iii) Sort frequent items in decreasing order based on their support. Use this order when building the FP-Tree, so common prefixes can be shared.

Pass 2:
Nodes correspond to items and have a counter
(i) FP-Growth reads 1 transaction at a time and maps it to a path
(ii)Fixed order is used, so paths can overlap when transactions share items In this case, counters are incremented
(iii)Pointers are maintained between nodes containing the same item, creating singly linked lists (dotted lines)
(iv)Frequent item sets extracted from the FP-Tree.

Frequent Itemset Generation
(i)Each prefix path sub-tree is processed recursively to extract the frequent itemsets. Solutions are then merged.
Conditional FP-Tree
The FP-Tree that would be built if we only consider transactions containing a particular itemset (and then removing that itemset from all transactions).

- Advantages of FP-Growth
  - only 2 passes over data-set
  - "compresses" data-set
  - no candidate generation
  - much faster than Apriori
- Disadvantages of FP-Growth
  - FP-Tree may not fit in memory
  - FP-Tree is expensive to build

V. DISTRIBUTED ALGORIHM

Distributed algorithm can be divided into two categories. Data distribution algorithm (DD) and the Count Distributed algorithm (CD).The basic idea of DD algorithm is that the candidate set evenly distributed among the nodes. The increasing traffic between the nodes can reduce the efficiency of this algorithm. CD algorithm is a typical parallel algorithm based on apriori algorithm. Each sub database scans the data base to calculate the support of each candidate set and then set the sum of the entire candidate set as the global support .If the global support is greater than the min_support ,we consider the itemset as the global frequent item. Also increase the node, that time traffic also increase and then efficiency also decrease.

ALGORITHM

In this paper we assume database contains 3 nodes S1,S2,S3. The database is divided into DB1, DB2 and DB3. we assume that the minimum support s=20%.Then generate local frequent itemset through the distributed algorithm(DA) from DB1,DB2 and DB3.Then all the local frequent item set are sent to the common node S in order to generate global frequent itemset.

Here we are having 4 cases

1.If a local frequent itemset occurs in all databases then that local frequent itemset will be a global frequent itemset.

2. If an itemset (not local frequent itemset) whose support value satisfies the minimum support value the it will be a global frequent itemset.

3.If an item set has 2 local supports value and if the sum less than the minimum support value it may be a global frequent itemset if it satisfies the foll.condn such that globalmin_support-current_minsupport < local_min_support(totcountonodes-cntoflocalsupport)

4.If an item set has 2 local supports value and if the sum less than the minimum support value it may not be a global frequent itemset if itdoes not satisfies the foll.condn such that globalmin_support-current_minsupport < local_min_support (totcountofnodes-cnttoflocalsupport)

Algorithm

Input:Transactional database,Support and Confidence

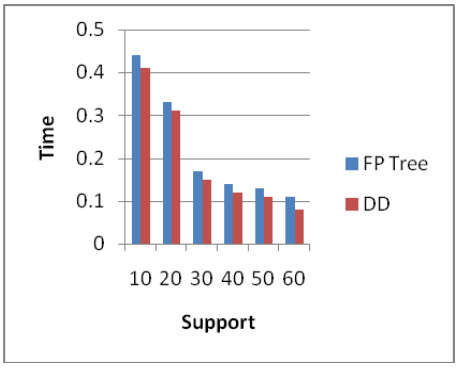
Output: Global frequent Itemset
1.Divide the Database into DB1,DB2 and DB3.
2.find the local frequent itemset using improved apriori algorithm
3.generate global frequent itemset

VI. EXPERIMENTAL RESULTS

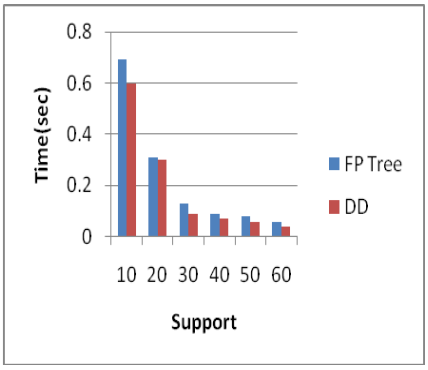
For conducting experimental result the proposed work has been implemented using connect data set. This distributed algorithm shows the accuracy by varying confidence and support values.

Sample results for DB1:

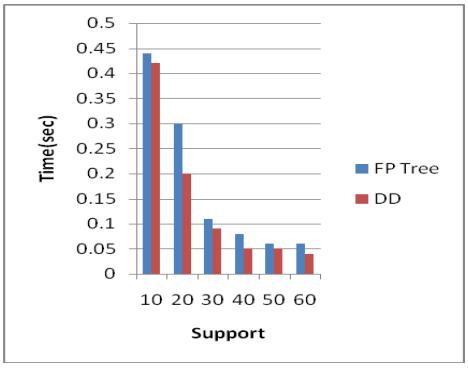
Accuracy for confidence=50



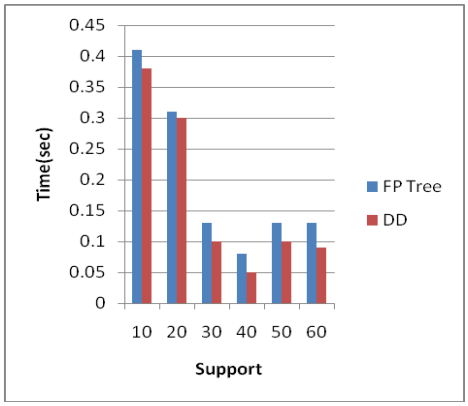
Accuracy for confidence=60



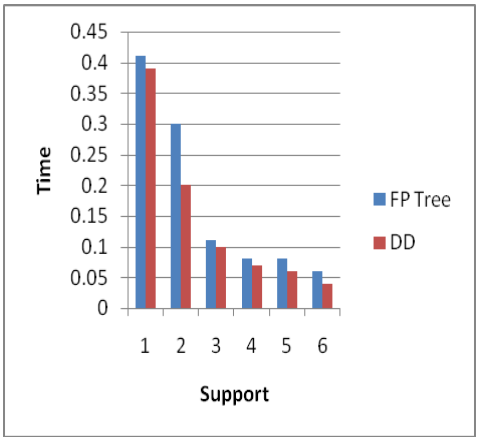
Accuracy for confidence=70



Accuracy for confidence=80



Accuracy for confidence=90



VII. CONCLUSIONS

Distributed database(DD) algorithm provide more accuracy, efficiency .If we compare this algorithm with FP-Tree this algorithm finds the global frequent itemset with more accuracy and efficiency. Thus the DD algorithm has been implemented and verified with the original dataset.

VIII. FUTURE ENHANCEMENT

By applying various technique like sampling technique this work can be extended in future and the efficiency and accuracy of the algorithm can be improved further.

REFERENCES

1.Lijuan Zhou,Shuang Li,Mingsheng Xu,"Research on Algorithm of Association Rules in Distributed Database System",2010,IEEE,2nd International Asia Conference on Informatics in Control, Automation and Robotics.

2.Rakesh Agrawal , Tomasz Imielinski , Arun Swami, "Mining association rules between sets of items in large databases", 1993, Proceedings of the ACM SIGMOD international conference on Management of data, p.207-216.

3.Cheung.D.W,"A fast distributed algorithm for mining association rules",4th International Conference on Parallel and Distributed Information Systems,1996,P31-42.

4. Thabet Slimani ," Efficient Analysis of Pattern and Association Rule Mining Approaches " IJITCS Vol. 6, No. 3, February 2014

