Frequent Itemset Discovery in E-commerce Domain: A novel Approach

Venkateswari S
Department of Software Engineering
Noorul Islam University, Kumaracoil,
K.K.Dist, India

Suresh R.M
Department of Computer Science
RMD Engineering College, Kavaraipettai,,
Chennai India

Abstract-In this paper we propose a structure to find the frequent itemsets from the e-commerce data. Frequent itemset discovery is a heavily researched area in the data mining field. Due to the rapid growth of online business culture, there are plenty of data available in e-commerce web servers. The proposed structure explains how data is collected, cleaned and mined to find frequent itemsets from the e-commerce domain. A new and efficient ILLT (Indexed Limited Level Tree) algorithm developed for discovering the frequent itemsets discussed in this paper. This algorithm can be established for mining large item sets or for any n transaction and also it is applicable for online and offline process. ILLT algorithm works in two phases. First the transactional data is converted into three level compact tree structures. Then this tree is scanned to discover the frequent itemsets for the given support level. ILLT algorithm determines the frequent item sets in the given database without doing multiple scans and extensive computations. The computing result shows that the introduced algorithm is producing the same output as the existing Apriori algorithm, and it can avoid missed mining effectively.

Key words: ILLT, Limited Level Tree, Frequent Itemset, E-Commerce, Data Mining

I. INTRODUCTION

Data mining proposes many solutions for the extraction of significantly and potentially useful patterns from large collection of data. One among them is association rule mining. Past few years discovering association rules is the focus of many studies. Finding frequent itemsets in the given transactions and finding association between itemsets are the two steps in association rule mining [1]. The first step, finding the frequent itemsets is very resource consuming task and it is the popular research field in data mining. Many algorithms are proposed to find the frequent itemsets[2][3][8][9]. These algorithms work on large volume of transaction data. E-commerce data are a best choice for this. Electronic commerce is now the killer domain for data mining technology[4][5]. Since e-commerce is growing fast and the data electronically collected are rich, clean, plenty and reliable, they are the optimal domain for mining[6][7]. Analyzing these data and adapting association rules, administrator can provide online customers information of great value, rule the market order and standardize the category, the price and the quality of various goods. We propose a structure in this paper to find the frequent itemsets from e-commerce data.

The remainder of the paper is organized as follows: The proposed structure is described in section 2. Experimental evaluation and performance study are discussed in section 3. Conclusion is presented in section 4.

II. PROPOSED STRUCTURE

Proposed structure for discovery of frequency itemsets from e-commerce data is given in figure 1. First module contains the transactional data from e-commerce domain. Next is the preprocessing module. Here the noise in the data is cleaned and transformed into a form which is suitable for mining. The third module is data mining module. This module uses the new and efficient ILLT algorithms for finding frequent itemsets.

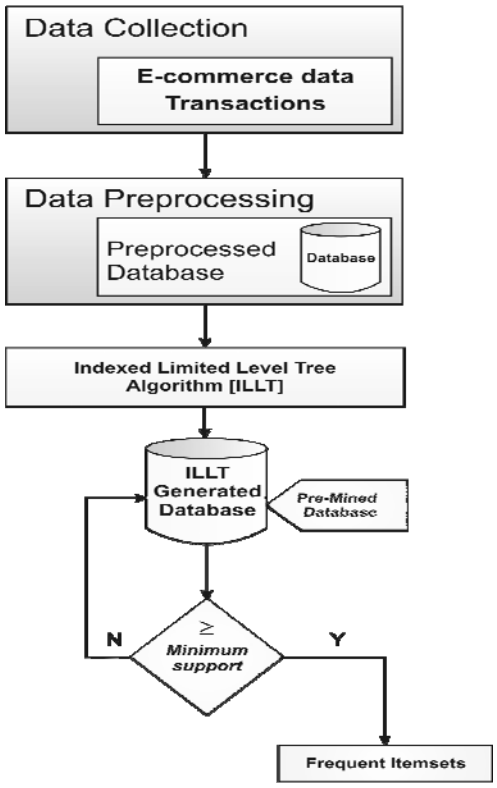

Figure 1. Proposed structure

A. E-commerce data

E-commerce data can be classified as web usage data, web content data , web structure data and user data. The web usage data is generated by interaction between person browsing the sire and servers of the e-commerce platform. These are stored as log files. The information available in web log files can say what prospective customers are seeking from a site. The data collected from these weblog files are rich containing information on prior purchase activity. The actions like what the customers look at, what they put into their shopping cart and do not buy and so on.

B. Data preprocessing

Data collected in the log files of the server cannot be used for mining purpose in the form as it is stored. So preprocessing is an essential activity before mining the data. Data preprocessing focus on four major aspects, removing records which are irrelevant for mining, user identification, and session identification and performing path completion. Preprocessing algorithms are appointed and the resultant data is obtained as tables suitable for the ILLT algorithm to find the frequent itemsets.

C. Data mining

The first step of association rule mining is the frequent itemset discovery. In our proposed structure, frequent itemsets are discovered using Indexed Limited Level Tree algorithm(ILLT). ILLT algorithm works in two phases. The first phase scans the given database and generates candidate itemsets for each transaction. These candidate itemsets are stored in a special compact tree structure called ILLTree. Once the tree is constructed there is no need for scanning the original database again. Second phase is frequent itemset generation. In this phase tree structure is explored to find frequent itemsets for different support levels.

D. ILLT Algorithm

The ILLT algorithm that creates the ILLTree structure is as follows.ALGORITHM : Indexed LimitedtLevelTree (Transaction)

Input : Minimum support

```
T: Transaction
Ci: Candidate set's of  T
FOR each element in Ci DO
    CandidateLength: Number of item count in Ci
     Look for the Tree with label CandidateLength for process
IF no Tree with label  CandidateLength  THEN
    Create a new Tree and label it as candidateLength  and alter all low level Tree Structure as the new Tree
    Select the tree with label CandidateLength for process
    Update the CandidateSet Ci in the selected Tree nodes
End IF
    If all Transaction over then
    FOR each Tree Ti DO
        FOR each CandidateSets Ci in the Tree  Ti
        DO
            IF CandidateSet Ci >= MinimumSupport THEN
            Add Ci to FrequentItems List
        END FOR
        Write the FrequentItems List
    END FOR
END IF
END FOR
```

E. The Process

First step of ILLT algorithm is construction of tree data structures. The levels of the tree are limited to three with an index node so it is named as Index Limited Level Tree (ILLT). The compact trees constructed in the first step is carried out by doing only one scan of given transactional database. From the resultant ILLTree it is easy to find frequent itemsets for different support levels. Scanning the database again is not needed at any stage. The tree structures store the contents of the transactions in their nodes.

ILLT algorithm scans the database first and generates candidate itemsets for every transaction in the database. These generated candidate itemsets are of different lengths. An unique tree is constructed for each k-length itemsets. Level 1 of the trees is the header node. This header node with the label H indicates which k- itemset that tree stores. If the label of the header node is 3, then this tree stores all 3-itemsets. The header node has *n* children at level two. *n* is the total number of items that occur in the transactions. Each node in the level two indicates an individual item. Third level of the tree stores the candidate itemsets as its nodes. An index is associated with each tree. The Indexed Limited Level Tree structure with three levels is shown in the figure 2.

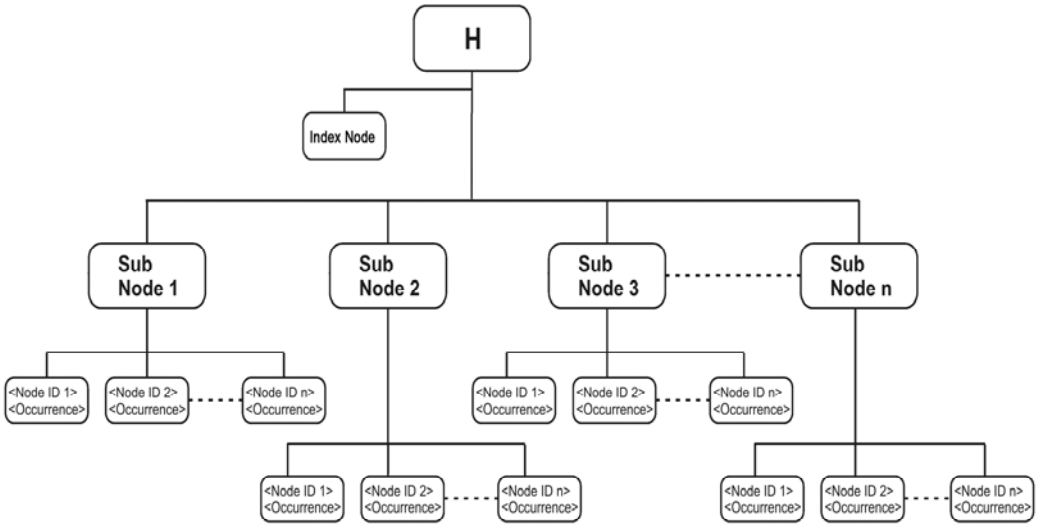

Figure 2. ILLTree structure

When candidate itemsets are generated, an unique identification number is assigned to each of them. This candidate itemset with its identification number is stored in the index first. Then this itemset is stored in the third level of the tree in the following syntax.

<Node Id number><Number of occurrence>

When a new candidate itemsets has to be inserted in the tree, it is first checked with the index. If it already exists in the index, only the occurrence number is incremented in the respective nodes. If it does not exist, then a new identification number is assigned and the itemset is stored in the tree. Depending on the length of the candidate itemset the appropriate tree is chosen for insertion.

ILLT algorithm thus constructs the tree structure for the given database. The entire database is stored as compact three level tree structure. Frequent itemsets are discovered by scanning this data structure. To find the frequent 1-itemsets the tree with the header label 1 is searched. Similarly for the frequent k-itemsets the tree with header label k is searched. Frequent itemsets for various support levels are achieved by comparing the support level with the occurrence number in the third level node. If the occurrence value is greater than or equal to the given support level then the items are frequent ones. Thus ILLTtree structure provides the facility of discovering frequent itemsets for any support levels at any time without scanning the database again.

III. EXPERIMENTAL EVALUATION AND PERFORMANCE STUDY

Experiments are conducted to test the ILLT Algorithm by comparing it with Apriori[11]. Program was written in Java and performed on a Pentium core 2 duo 2.8GHz PC with 2GB RAM running on windows XP.

Datasets are generated using synthetic data generation program [10] of the IBM Almaden Quest research group available at: http://fimi.cs.helsinki.fi.data/T. T10I4D100k dataset are used. The Both Apriori and ILLT algorithm are run over the same data set and the frequent itemsets found by the two algorithms are the same.

Figure 3 shows the execution time in seconds for various datasets with transaction sizes 50, 100, 150 till 2000 for both Apriori and ILLT algorithms. From the figure , we can conclude easily that ILLT algorithm performs better than Apriori.

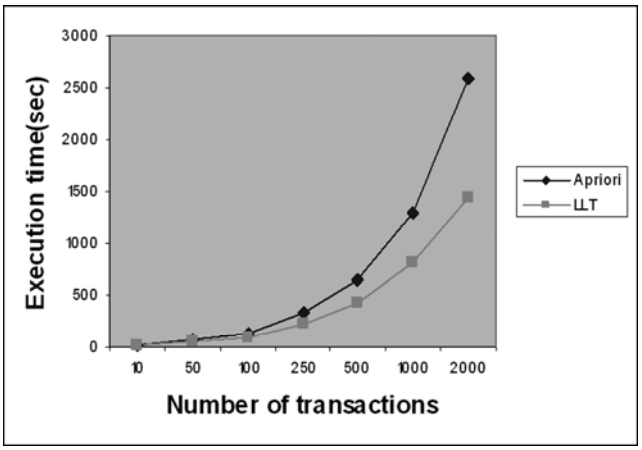

Figure 3

The memory requirement of ILLT algorithm depends only on the number of distinct items in the database. So it is independent from the size of the database. So the memory requirement of Apriori algorithm comparing with ILLT algorithm is high.

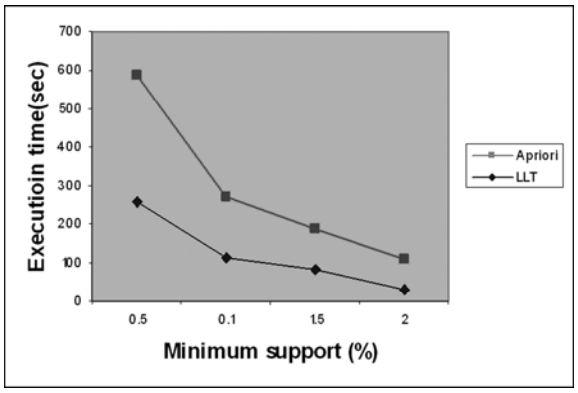

Figure 4

When the minimum support increases the number of frequent itemsets decreases. Figure 4 shows the execution time comparison of Apriori and ILLT algorithms. It is clear from the Figure that ILLT algorithm performs better than Apriori algorithm for all minimum support values.

IV. CONCLUSION

In this paper, we have proposed a structured approach that can discover frequent itemsets from e-commerce data. E-commerce transactions are collected from web log files and they are preprocessed using preprocessing algorithms before mining. Preprocessed data are mined using an efficient algorithm ILLT. This algorithm finds the frequent itemsets for any given support level . Experimental results shows that ILLT algorithm overcomes the multi-scan problem and extensive computation involved in Apriori and other traditional algorithms. Since the entire database need not be brought in to memory, memory management is efficient in ILLT algorithm. This structured approach helps to predict the purchase behavior of the on line purchasers which can be utilized to improve the e-commerce website and e-business.

REFERENCES

[1] J. Han and M. Kamber, Data Mining: Concepts and Techniques. Morgan Kaufman, San Francisco, CA, 2001.
[2] J. Yu, Z. Chong, H. Lu, and A. Zhou. “False Positive or False Negative: Mining Frequent Itemsets from High Speed Transactional Data Streams.” In Proc. of VLDB, 2004.
[3] Li Y. C., Yeh J. S., Chang, C. C.: “ Direct candidates generation: a novel algorithm for discovering complete share-frequent itemsets.” In Proceedings of the 2nd Intl.Conf. on Fuzzy Systems and Knowledge Discovery, pp. 551–560, 2005.
[4] Ansari S,Kohavi R, Mason L, Zheng Z 2001 “Integrating e – commerce and data mining: architecture and challenges.” In Proc. 2001 IEEE Int. Conf. on Data Mining (New York: IEEE Comput. Soc.) pp 27 – 34.

[5] Kohavi R 2001 “Mining e – commerce data: The good, the bad, and the ugly.” In proceedings of the Seventh ACMSIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2001) (New York: ACM press) pp. 8 – 13.
[6] N R Srinivasa Raghavan, “Data Mining in e- commerce: A servey”,Sadhana, vol, 30, n0.2, 2005, pp. 275 – 289
[7] Yuantao Jiang and Siqin Yu, “Mining the E-commerce Data to Analyze the Target Customer Behaviour”, workshop on knowledge discovery and Data Mining, pp. 406-409,2008.
[8] R. Agarwal, C. Aggarwal, and V. V. V. Prasad. “A tree projection algorithm for generation of frequent itemsets.” In J. of Parallel and Distributed Computing AIML Journal, Volume (6), Issue (3), September, 200659 (Special Issue on High Performance Data Mining), 2000.
[9] Tang, P., Turkia, M., “Parallelizing frequent itemset mining with FP-trees. “ Technical Report. titus.compsci.ualr.edu/~ptang/papers/par-fi.pdf, Department of Computer Science, University of Arkansas at Little Rock, 2005.
[10] IBM. Almaden. Quest synthetic data generation code. http://fimi.cs.helsinki.fi.data/T.
[11] R. Agrawal and R. Srikant. “Fast algorithms for mining association rules.” In Proc. 1994 Int. Conf. Very Large Data Bases, pages 487-499, Santiago, Chile, September 1994.