Mining Positive and Negative Association Rules Using FII-Tree

T Ramakrishnudu
Dept. of Computer Science and Engineering
National Institute of Technology
Warangal, AP, India

R B V Sbramanyam
Dept. of Computer Science and Engineering
National Institute of Technology
Warangal, AP, India

***Abstract*—Positive and negative association rules are important to find useful information hidden in large datasets, especially negative association rules can reflect mutually exclusive correlation among items. Association rule mining among frequent items has been extensively studied in data mining research. However, in recent years, there has been an increasing demand for mining the infrequent items. In this paper, we propose a tree based approach to store both frequent and infrequent itemsets to mine both the positive and negative association rules from frequent and infrequent itemsets. It minimizes I/O overhead by scanning the database only once. The performance study shows that the proposed method is an efficient than the previously proposed method.**

Keywords—data mining; association rule; frequent itemset; positive association rule; negative association rule

I. INTRODUCTION

Association rule mining is a data mining task that discovers associations among items in a transactional database. Association rules have been extensively studied in the literature since Agrawal et al. first introduced it in [1, 2]. A typical example of association rule mining application is the market basket analysis. Much effort has been devoted and algorithms proposed for efficiently discovering association rules [2, 3, 4, 5, 6,16].

Association rules provide a convenient and effective way to identify and represent certain dependencies between attributes in a database[1]. Association rule mining includes positive and negative association rule mining[9,11,12,17]. In the traditional approach to find association rules, one merely thinks in terms of positive association rules: especially when determining the degree of support and confidence[2].

The study of the negative association rule is a new active research field in recent years. It still focuses on the transactional databases, and has made a number of important research results[7,8,10,18]. Brin M. et al. referred to the relevance of the two sets firstly [1]. Savasere O. et al. described a strong negative association rules model [2]. Xindong Wu et al. proposed a PR model [3], and gave an algorithm that can mine positive and negative association rules simultaneously.

Ling Zhou et al. [14] and Junfeng Ding et al. [15] proposed methods to mine association rules from infrequent itemsets. Mining positive association rules from frequent itemsets and negative association rules from infrequent itemsets with some interesting measures are described in [9]. Honglei Zhu et al. [12] mine both positive and negative association rules from frequent and infrequent itemsets respectively with the differential support and confidence.

In this paper we examine the problem of mining positive and negative association rules from frequent and infrequent itemstes.

The rest of this paper is organized as follows. Section 2 briefly presents the relevant concepts and definitions. In Section 3, the existing strategies for mining both positive and negative association rules are reviewed. The proposed algorithm is presented in Section 4. Section 5, illustrate the computational results. The concluding remarks are finally made in Section 6.

II. CONCEPTS AND DEFINITIONS

Let $I = \{i_1, i_2, i_3 ... i_n\}$ be a finite set of items and *DB* be a transactional database. Support of the itemset $X \subseteq I$ is [1]:

Supp(X) = No.of transactions contains X/Total No.of Transactions in DB (1)

Definition 1: If the support of itemset *X* is greater than or equal to user defined minimum support (*ms*) threshold, *X* is called frequent itemset otherwise infrequent Itemset [5].

A. Positive Association Rule:

A (positive) association rule is of the form: $X \Rightarrow Y$, with $X, Y \subseteq I$, $X \cap Y = \emptyset$[1][9]. Support and confidence of $X \Rightarrow Y$ are defined as[2]:

$$Supp(X \Rightarrow Y) = Supp(X \cup Y) \quad (2)$$

$$Conf(X \Rightarrow Y) = \frac{Supp(X \cup Y)}{Supp(X)} \quad (3)$$

An interesting positive association rule has support and confidence greater than user given thresholds *minimum support (ms)* and *minimum confidence (mc)* respectively.

B. Negative Association Rule:

A negative association rule is an implication of the form $X \Rightarrow \neg Y$ (or $\neg X \Rightarrow Y$), where $X \subseteq I$, $Y \subseteq I$, and $X \cap Y = \emptyset$[9]. The rule $\neg X \Rightarrow \neg Y$ is equivalent to a positive association rule in the form of $Y \Rightarrow X$. From [12], we extracted the following formulas:

$$Supp(\neg X) = 1 - Supp(X) \quad (4)$$

$$Supp(\neg X \cup Y) = Supp(X) - Supp(XUY) \quad (5)$$

$$Supp(X \cup \neg Y) = Supp(Y) - Supp(XUY) \quad (6)$$

$$Supp(\neg X \cup \neg Y) = Supp(X) - Supp(Y) + Supp(XUY) \quad (7)$$

$$Conf(X \Rightarrow \neg Y) = \frac{Supp(X) - Supp(X \cup Y)}{Supp(X)} \quad (8)$$

$$Conf(\neg X \Rightarrow Y) = \frac{Supp(X) - Supp(X \cup Y)}{1 - Supp(X)} \quad (9)$$

$$Conf(\neg X \Rightarrow \neg Y) = \frac{1 - Supp(X) - Supp(Y) + Supp(X \cup Y)}{1 - Supp(X)} \quad (10)$$

$$Corr(X,Y) = \frac{Supp(X \cup Y)}{Supp(X)Supp(Y)} \quad (11)$$

Definition 2: The bit vector *(BV)* of an item *i* is in form *BV= (b_1, b_2, b_3 ...bm)*, $b_k \in [0, 1]$, If $i \in T_k$, and then b_k=1, otherwise b_k=0, *where k=1, 2 ...m.* The size of BV is equal to the number of items in *i*, and the support of an itemset is equal to the number 1s in the bit vector.

III. Related Work

Several algorithms have been proposed for mining association rules, negative association rules. But only few a algorithms have been proposed for mining both positive and negative association rules concurrently. Wu et al [9] presented an Apriori-based framework for mining both positive and negative ARs based on rule dependency measures and an additional threshold *minimum interest (mi).* A rule $X \Rightarrow \neg Y$ (or $\neg X \Rightarrow Y$) is only considered as a valid negative AR, if both *X* and *Y* are frequent and the *interest (X, ¬Y) ≥ mi (*or *interest (¬X, Y) ≥ mi)*.

The most common frame-work in the association rule generation is the "Support-Confidence" one. In [11], authors considered another frame-work called correlation analysis that adds to the support-confidence. They combined the two phases (mining frequent itemsets and generating strong association rules) and generated the relevant rules while analyzing the correlations within each candidate itemset. Their algorithm avoids evaluating item combinations redundantly. For each candidate itemset, they computed all possible combinations of items are outputted to analyze their correlations. At the end, they keep only those rules generated from item combinations with strong correlation. If the correlation is positive, a positive rule is discovered. If the correlation is negative, two negative rules are discovered.

Honglei Zhu et al. [12] proposed for the purpose of simultaneously generating positive ARs from frequent itemsets and negative ARs from infrequent itemsets with differential minimum support and differential minimum confidence. An innovative approach has proposed in [13]. In this, authors dividing the itemset space into four parts for mining positive and negative association rules. In [14], the authors proposed a method to mine association rules form infrequent itemsets.

IV. Problem Description And Proposed Method

Most of the methods proposed for mining positive and negative association rules, maintains both frequent and infrequent itemsets and hence suffer from scalability. To maintain the execution time within user's expectations, it is necessary to design an efficient approach to mine both positive and negative association rules.

Problem Statement: Given a database of transactions DB and user-defined minimum support (ms) value, minimum confidence (mc) values, the problem is to extract all interesting positive and negative Boolean association rules.

We propose Frequent and Infrequent Itemset tree (FII-tree) as a data structure to hold requisite itemsets and also a method to extract all the positive and negative association rules.

The proposed process consists of two phases.

Phase 1: In this phase, we construct an FII-tree which can hold all frequent and infrequent itemsets. The root of the tree, labeled with "null". Each non-root node in a tree has generic form *<I, c>*, where *I* is an itemset and *c* is the support of I. Infrequent 1-itemsets are ignored. However, all frequent 1-itemsets are placed in level 1 in lexicographic order, all frequent and infrequent 2-itemsets are in level 2, and so on, all frequent and infrequent k-itemdsets are placed in level k. The highest level of the FII-tree is *L*, where *L* is equal to the number of frequent 1-itemsets. For FII-tree, two indices, one is *FreqIndex* for all frequent itemset lists and the second one is *InfreqIndex* for all infrequent itemset lists are maintained separately for easy accessibility of frequent and infrequent itemsets. The step by step process of Creation of FII-tree is given below:

1) Scan the database DB once, and store in an item based vectors BV, then find frequent 1-itemsets based on definition2.

2) Insert frequent 1-items one by one in the tree, and assign to an index FreqIndex

3) Generate candidate k-itemsets C_k (k=2, 3 ...) from frequent (k-1)-itemsets. For each item X in a candidate k-itemsets C_k

a) If supp(X) ≥ min_supp and Corr(X)>1 then assign X to frequent k- itemset list (FL_k) otherwise assign X to infrequent k-itemset list (IFL_k). Calculate support of X by performing bitwise AND operation between bit vectors (BV) (if $x_1, x_2 \in X$ then the supp(X) = $x_1 \wedge x_2$).

b) Assign the FL_k and IFL_k to $FreqIndex_k$ and $InfreqIndex_k$ respectively (where k= 2, 3, 4…).

4) Repeat step3 until to generate largest itemset.

Phase 2: Mining Positive and Negative Association Rules:

5) Read frequent k-itemsets (k=1, 2, 3…) from FII-tree and generate Positive Association Rules based on given threshold values.

6) Read frequent k-itemsets (k=1, 2, 3…) from FII-tree and generate Positive Association Rules based on given threshold values.

Example: Transactional Database (DB):

TABLE I. TRANSACTIONAL DATABASE

TID	Items	TID	Items
1	A B C D E	6	A B C
2	A B C	7	A B C
3	A B D	8	B C E
4	B C D	9	B C D
5	C D E	10	C D

The Transactional Database (DB) consists of 5 items and 10 transactions. The Item based bit vectors are:

BV1 = 1110011000
BV2 = 1111011110
BV3 = 1101111111
BV4 = 1011100011
BV5 = 1000100100

Based on definition2 the support count of an itemset is the number of 1s in each bit vector. For example *Supp(A)=5, Supp(B)=8, Supp(C)=9 Supp(D)=6 and Supp(E)=3.*The *minimum support (ms) is 5 (50%),* then the frequent 1-itemsets are: {A, B, C, D}.

Insert frequent 1-itmsets in FII-tree. The candidate 2-itemsets are AB, AC, AD, BC, BD and CD. Next perform the bitwise AND (^) operation between each pair of frequent itemset.

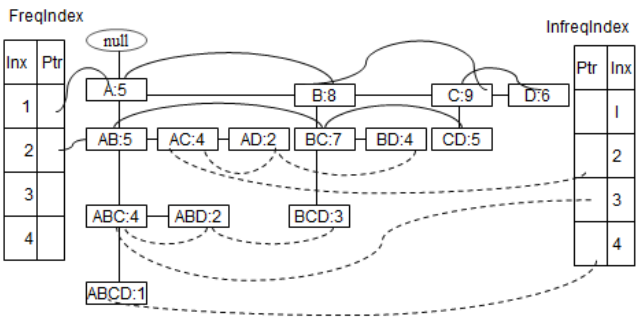

Fig. 1. FII-tree

For example, R=BV_1^ BV_2= (1110011000) ^ (1111011110) =1110011000, the number of 1s in the resultant bit vector R is 5, so the *Supp(AB)=5, Supp(AC)=4, Supp(AD)=2, Supp(BC)=7, Supp(BD)=4 and Supp(CD)=5.* Thus the frequent 2-itemsets are {AB, BC, and CD} as their support counts are not less than 50%, and the infrequent 2-itemsets are {AC, AD, and BD} as their support counts are less than 50%.

Insert frequent and infrequent 2-itemsets in FII-tree. For three itemsets perform the bitwise AND operation between three bit vectors as given below: R = (BV_1 ^ BV_2 ^ BV_3) = (1110011000) ^ (1111011110) ^ (1101111111) =1100011000, the number of 1's in the resultant bit vector is 4, so the *Supp(ABC)=4, Supp(ABD)=2, and Supp(BCD)=3.* Thus the infrequent 3-itemsets are {ABC, ABD, and BCD} as their supports are less than 50% and there are no frequent 3-itemsets and 4-itemsets, *Supp(ABCD)*=01, this is the largest itemset for the given transactional database (DB), then the algorithm stop processing. The FII-tree for the above example is shown in fig1.

In the Fig. 1, solid straight lines are the links between the items in the same levels and links between different levels of the tree. All arcs are the links between frequent itemsets and the dashed arcs are the links between infrequent items. Different levels of frequent items are linked to frequent index called *FreqIndex* and infrequent items are linked to infrequent index called *InfreqIndex.*

Algorithm 1:
Input: MinSup, MinConf.
Output: FrequentItemset, InfrequentItemset, PositiveAR and NegativeAR.

Phase 1:

1) Scan database DB once and find frequent 1-itemset, $freq_1$

2) Insert $freq_1$ in FII-tree: Insert_FII-tree () //calls algorithm2

3) Generate candidate k itemset C_k (k=2, 3...) from $freq_1$ and Insert $freq_k$ in FII-tree: Insert_FII-tree()

Phase 2:

// *FreqIndex* frequent itemset index
// *InfreqIndex* infrequent itemset index.

4) Generate Positive Association Rules from $FreqIndex_k$ (k=1, 2…)

If Corr(X, Y)>1 then
If Conf(X, Y)> MinConf then
PositiveAR← PositveAR U {X→Y}

5) Generate Negative Association Rules from $InfreqIndex_k$ (k=2, 3…)

For each infrequent item i in *InfreqIndex*
For each expression X Y, X U Y = i and X ∩ Y =Ø
{
Step A:
If Corr(X,Y)<1 then
If Supp(X, ¬Y)>MinSupp and Conf(X, ¬Y)>minconf then
NegativeAR←NegativeAR U (X→¬Y)
Step B:
If Corr(X,Y)<1 then
If Supp (¬X, Y)>MinSupp and
Conf (¬X, Y)>minconf then
NegativeAR←NegativeAR U (¬X→Y)
}

6) AR← PositivevAR U NegativeAR.

Algorithm 2:
Input: MinSup, Candidate Itemset Ck, k
Output: Frequent Inferquent Index Tree (FII-tree)

```
Inseret_FII-tree ( )
{       // k=1 for frequent 1-itemsets
        //k= 2, 3… for frequent 2, 3..
1: if k==1 then
   {
```

For all items in $freq_1$
Insert i, i∈$freq_1$ in a node then add node to the tree.
$FreqIndex_1$← $FreqL_1$
//$FreqL_1$ is frequent itemset list1
}
2: else{
for all items in Ck
{
//$FreqL_k$ is k frequent itemset list
//$InFreqL_k$ is k infrequent itemset list
Insert i, i ∈C_k in a node then add node to the tree.
if supp(i)> MinSup then
$FreqL_k$← i
else $InFreqL_k$← i
$FreqIndex_k$ ← $FreqL_k$
$InfreqIndex_k$ ← $InFreqL_k$
}
}

V. Experimental Results

We conduct experiments on a different transaction size and differing number of transactions in a database to compare our approach with the PNAR [12]. The execution time with different minimum supports for the dataset T50I30D200K is shown in the Fig. 2.

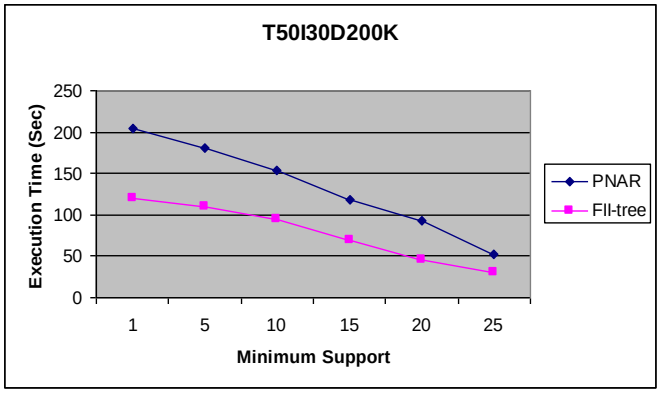

Fig. 2. Execution time with different minimum supports

The execution time with different dataset sizes (number of transactions) for the fixed minimum support 0.5 is shown in Fig. 3. It can be observed that both the methods generate equal number of positive and negative association rules, but the proposed approach reduce the execution time over the existing method.

VI. Conclusion And Future Work

In this paper, we have designed a new tree structure to store both the frequent and infrequent itemsets for mining both positive and negative association rules. In the proposed method the database is scanned only once for mining positive and negative association rules, so it reduces the number of I/O operations. Another flexibility of the structure is, if any new frequent 1-itemsets are mined by reducing the user threshold value (minimum support (ms)), the proposed method allows appending of new items to the tree without reconstructing from scratch.

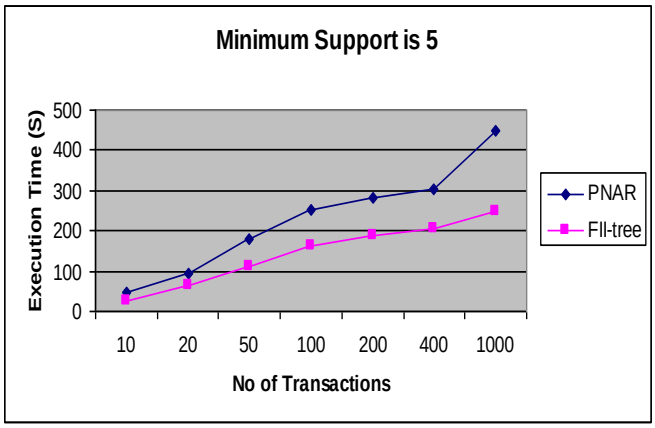

Fig. 3. Execution time with different dataset size

Recently, there have been some interesting studies about mining frequent patterns in databases which allow adding new data or deleting old data. The maintenance of the already mined frequent patterns when updating databases is an interesting topic for future research.

References

[1] R. Agrawal, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases", In Proceedings of ACM SIGOM International Conference on Management of Data, New York, May 1993, pp. 207–216.

[2] R.Agrawal and R.Srikant, "Fast algorithms for mining association rules", In Proceedings of 20th International Conference on VLDB, Chile, May 1993, pp. 207–216.

[3] J.Han and Y.Fu, "Mining multiple-level association rules in large databases", IEEE Trans. on Knowledge and Data Engineering, Vol. 11, No 5, September 1999, pp. 798-805.

[4] M.Klemettinen, H.Mannila, P.Ronkainen, H.Toivonen and A.I.Verkamo, "Finding interesting rules from large sets of discovered association rules", In proceedings of 3rd International Conference on Information and Knowledge Management, Maryland, November 1994, pp 401-407.

[5] R.Srikant and R.Agrawal, "Mining quantitative association rules in large relational tables", ACM SIGMOD Record, Vol. 25, No. 2, June 1996, pp.1-12.

[6] Brin.S, Motwani.R, and Silverstein.C, "Beyond Market: Generalizing association rules to correlations", In Proceedings of ACM SIGMOD International Conference on Management of Data, New York, May 1997, pp. 265-276.

[7] Savasere.A, Omiecinski.E, and Navathe.S, "Mining for strong negative associations in a large database of customer transactions", In Proceedings of 14th International Conference on Data Engineering, Orlando, February 1998, pp. 494-502.

[8] X. Yuan, B.P. Buckles, Z. Yuan and J. Zhang,"Mining negative association rules", In Proceedings of 7th International Symposium on Computers and Communication, Italy, July 2002, pp. 623-629.

[9] X. Wu, C. Zhang and S. Zhang, "Efficient mining of both positive and negative association rules", ACM Trans. on Information Systems, Vol.22 No. 3, July 2004, pp. 381-405.

[10] S. Wang, H. Song, and Y. Lu, "Study on negative association rules," Trans. of Beijing Institute of Technology, Vol. 24, No. 11, November 2004, pp. 978-981.

[11] M.L. Antonie and O.R. Za¨ıane,"Mining positive and negative association rules: an approach for confined rules", In Proceedings of International Conference on Principles and Practice of Knowledge Discovery in Databases, Italy, September 2004, pp. 27-38.

[12] Honglei Zhu, Zhigang Xu, "An Effective Algorithm for Mining Positive and Negative Association Rules", In Proceedings of International

Conference on Computer Science and Software Engineering, Hubei, December 2008, pp. 455-458.

[13] Chris Cornelis, peng Yan, Xing Zhang, Guoqing Chen, "Mining Positive and Negative Association Rules from Large Databases", In Proceedings of IEEE Conference on Cybernetics and Intelligent Systems, Bangkok, June 2006, pp. 1-6.

[14] Ling Zhou, Stephen Yau, "Efficient association rule mining among both frequent and infrequent items", Computers and Mathematics with Applications, Vol. 54, No.6, September 2007, pp. 737–749.

[15] Junfeng Ding, Stephen S.T. Yau, "TCOM, an innovative data structure for mining association rules among infrequent items", Computers and Mathematics with Applications, Vol. 57, No. 2, January 2009, pp. 290-301.

[16] Tamanna Siddiqui, M Afshar Alam and Sapna Jain, "Discovery of scalable association rules from large set of multidimensional Quantitative datasets", Journal of Advances in Information Technology, Vol. 3, No. 1, February 2012, pp. 69-76.

[17] Nikky Suryawanshi, Susheel Jain and Anurag Jain, "A review of negative and positive association rule mining with multiple constraint and correlation factor", International Journal of Emerging Technology and Advanced Engineering, Vol. 2, No. 12, December 2012, pp. 778-781.

[18] Li-Min, Shu-Jing Lin and Don-Lin Yang, "Efficient mining of generalized negative association rules", In proceedings of IEEE International Conference on Granular Computing, USA, August 2010, pp. 471-476.