A REVIEW ON ALGORITHMS FOR ASSOCIATION RULE MINING IN INTERTRANSACTION

Shital H. More[1], Swati A. Abhang[2]
[1,2] Assistant professor , Department of Information Technology
Thakur College of Engineering & Technology, Mumbai.

Abstract
Association rule discovery from large databases is one of the tedious tasks in data mining. Most of the previous studies on mining association rules are on mining intratransaction associations, i.e., the associations among items within the same transaction where the notion of the transaction could be the items bought by the same customer, the events happened on the same day, etc. Mining intertransaction associations has more challenges on efficient processing than mining intratransaction associations because the number of potential association rules becomes extremely large after the boundary of transactions is broken.
In this paper, we introduce the notion of intertransaction association rule mining given by algorithms such as EH-Apriori and FITI. FTTI generates many unnecessary combinations of items because the set of extended items is much larger than the set of items. Thus, In order to provide efficient solution we propose an approach of group-based intertransaction association rule mining, where this group of transactions are following certain constraints. This proposed solution will help in analyzing the predictions on stock market data.
Index Terms: **Association rule mining, EH-Apriori, FITI, Group-based Transactions etc.**

1. INTRODUCTION

A. ASSOCIATION RULE MINING

Association rule mining, the most important and well researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases.
Association rule mining find out association rules that satisfy the predefined minimum support and confidence from a given database. The problem can be decomposed into two subproblems. First to find those itemsets whose occurrences exceed a predefined threshold in the database; those itemsets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence [5].
Let, one of the large Itemsets is Tk, Tk = {I1, I2, … , Ik}, association rules with this itemsets are generated in the following way: the first rule is {I1, I2, … , Ik-1}⇒ {Ik}, by checking the confidence this rule can be determined. Then other rules are generated by deleting the last items in the antecedent and inserting it to the consequent, further the confidences of the new rules are checked to determine the interestingness of them. This process is iterated until the antecedent becomes empty.
In this paper, we will review algorithms for intertransaction association rule mining task. We will also investigate the application of Intertransaction association rules mining in stock price predication and the possibility of generalizing this method to futures market.
The remainder of this paper is organized as follows: In Section 2, we describe the problem of Intertransaction association rule mining in general. In Section 3, we will discuss EH-Apriori traditional approach, in Section 4 will have a detailed look over FITI algorithm. In addition to this we are also discussing a new approach of group-based intertransaction association rule mining.

2. INTERTRANSACTION ASSOCIATION RULE MINING

Anthony has explored the problem of intertransaction association rule mining [1]. The difference between the traditional association rule for intratransaction and intertransaction association rule can be stated as the following:

R2: ''When the prices of A and B go up, the price of C will increase on the same day with probability of 80%.''

However, stockjobbers may be more interested in the following rule.

R3: ''If the prices of A and B go up on the first day, the price of C will increase four days later with probability of 80%.''

Above classical association rules, like R2, discover the relationship among items within the same transactions, called as Intratransaction while R3 expresses association among items of different transactions along certain dimension, called Intertransaction.

2.1. PROBLEM DESCRIPTION:

Let B = {t1, t2 ... tn} be a transaction database, and each transaction is a set of items. [1] Tung et al. used the sliding window and extended-items to describe the intertransaction (see Definition 1) [1].

Definition 1. A sliding window W in a transaction database B is a block of w continuous intervals along domain D, starting from interval d0 such that B contains a transaction at interval d0. Each interval dj in W is called a subwindow of W denoted as W[j], where j= dj -- d0. We call j the subwindow number of dj within W.

Each sliding window W can be viewed as a continuous ω (a fixed interval called maxspan, or sliding_window_length) sub-windows such that each sub-window contains only one transaction. Let ei be an item, its occurrences in different transactions in a sliding window can be extended from ei(0) to ei(ω), where 0, ω are positions of transactions in the window. The transactions in a sliding window W can be merged into a megatransaction (or extended transaction) by putting all of W's extended items in a collection. Hence, an inter itemset refers to a set of extended-items, and an inter association rule can be represented as $X \rightarrow Y$, where X and Y are both a set of extended-items and $X \cap Y = \emptyset$.The definition of the support and confidence in inter association mining follows up the intra association mining. Let N be the number of megatransactions and, X and Y both be a set of extended-items and $X \cap Y = \emptyset$. Let Txy be the set of megatransactions that contains X and also Y, and Tx be the set of megatransactions that contains X [3].

We have,

$$sup(X \rightarrow Y) = |Txy| / N,$$

$$conf(X \rightarrow Y) = |Txy| / |Tx| \quad [3]$$

Example Let, the five transactions are located at intervals 1, 2, 4, 5, 6. Let w=4, we now have five sliding windows W1, W2, W3, W4 and W5, with addresses of 1, 2, 4, 5, 6, respectively. Each window contains 4 subwindows. For example, W1 has four subwindows W1[0] (with items a, b), W1[1] (with items b, d), W1[2] and W1[3] (with items a, b, c, d). Each sliding window forms a megatransaction, which is the itemset of all the items in one sliding window. In our case, the megatransaction in W1 is {a[0], b[0], b[1], d[1], a[3],b[3],c[3],d[3]}. Thus, we have, Σ ={ a1[0], b1[0], b1[1], d1[1], a1[3], b1[3], c1[3], d1[3], b2[0], d2[0], a2[2], b2[2], c2[2], d2[2], b2[3], c2 [3], a3[0], b3[0], c3[0], d3[0], b3[1], c3[1], a3[2], b4[0], c4[0], a4[1], a5[0]}. Then, after we set the two essential parameters minsup (denotes minimum support level) and minconf (denotes minimum confidence level), we can mine intertransaction association rules from the transaction database. For instance, setting minsup=0.4 and minconf=0.8, we can get one rule mined from Table 1 (see Figure 1),

{b[0], d[0]} ⟹ a[2] (support=0.4, confidence=1) [2].

TABLE I. TRANSACTION DATABASE

D	E
1	a, b
2	b, d
3	
4	a, b, c, d
5	b, c
6	a
7	
8	
9	

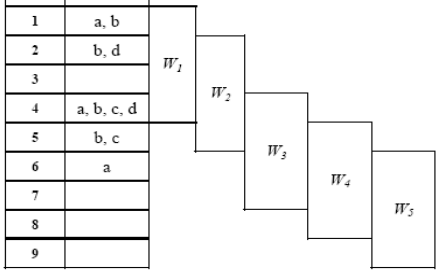

Figure 1. Mining Intertransaction Association Rules[2]

1. EH-APRIORI

To discover frequent intertransaction itemesets with the existence of a megatransaction within each sliding window leads to use the extended Apriori algorithm. Apriori algorithm assumes

the existence of lexicographic order among the extended-items, the extended-items in the megatransaction will be ordered using Definition 1 [1].

Definition 1. Let ei (di) and ej (dj) be two extended-items in a megatransaction. We say that ei (di) < ej (dj) if either of the following two conditions holds:

1. di < dj
2. di = dj and ei < ej.

Otherwise, we say that ej (dj) < ei (di) [1].

By using this definition, the Apriori algorithm is used to discover frequent intertransaction itemsets. To enhance the efficiency further, a hashing technique similar to the one in is used.

The general idea of the algorithm is as follows:

When the support of candidate intertransaction one itemsets is counted by scanning the database, information about candidate intertransaction two-itemsets is collected in advance in such a way that all the possible two-itemsets are hashed to a hash table. Each bucket in the hash table consists of a number to represent how many itemsets have been hashed to this bucket so far. The hash table is then used to reduce the number of candidate intertransaction two-itemsets. This is done by removing a candidate two itemsets if its corresponding bucket value in the hash table is less than minsup. We call this algorithm the Extended Hash Apriori or EH-Apriori [3].

2. FITI [FIRST INTRA THEN INTER] ALGORITHM

Unlike EH-Apriori which is modified from the Apriori algorithm, FITI is an algorithm designed specifically for discovering frequent intertransaction itemsets [7].

The problem of mining intertrasaction association rules can be decomposed as follows:

3. Find all Inter-transaction itemsets with support higher than minsup. We are calling these as Frequent Inter-transaction itemsets
4. For every frequent Inter-transaction itemset F and for all possible combination of X C F, output a rule
   $X \Longrightarrow (F - X)$ if its confidence is higher than minconf [7].

FITI makes use of the following property to enhance its efficiency in discovering frequent inter-transaction itemsets [1].

Property 1. Let F be a frequent intertransaction itemset.

Let, Ai = {ej |1 ≤ j ≤ u, ej(i) Є F }

For all i, 0 ≤ i ≤ (w-1) and Ai must be a frequent intratransaction itemset

Proof. We will prove this property by contradiction. Let F be a frequent intertransaction itemset.

Let Ai = {ej |1 ≤ j ≤ u, ej (i) Є F} for all, i, 0 ≤ i ≤ (w-1) Assume that ∃Ai, such that Ai is not a frequent intratransaction itemset. We denote the support of support.F. and support of Ai as support (Ai) Since F frequent intertransaction itemset, support(F) ≥ minsup. Also, since Ai is not a frequent intratransaction itemset, then support (Ai) < minsup.

Hence, we have

support(F) > support(Ai)

However, we know that for any sliding window W that contains F, Ai will occur in W[i] and each W[i] refers to a different transaction for different W. We can thus conclude that support(Ai) >= support[F] giving a contradiction and thus proving that Property 1 holds [1].

This property provides a different view of mining frequent intertransaction itemsets. Instead of viewing mining as an attempt to identify frequently occurring patterns formed from the extended items, we can view it as an attempt to discover frequently occurring patterns formed from frequent intratransaction itemsets.

As such in FITI, frequent intratransaction itemsets are first discovered and then frequent intertransaction itemsets are formed from them. This gives rise to the name of FITI which stands for First Intra Then Inter.

In FITI, in the first phase frequent intratransaction itemsets are discovered and stored in a data structure designed to facilitate the mining of frequent intertransaction itemsets in the later phase.

Each frequent intratransaction itemset is given a unique number called an ID. By using this ID as an index into the data structure, FITI is able to gather information on intratransaction itemsets quickly. To avoid the need to regenerate frequent intratransaction itemsets during the discovery of frequent intertransaction itemsets, the original database is transformed into another database that stores the IDs of frequent intratransaction itemsets presented in each transaction of the original database. When mining frequent intertransaction itemsets, each intertransaction itemset is represented as a tuple

of w IDs. Using this encoding, we formulate two types of joins to generate candidates intertransaction. (k+1)-itemsets from two existing frequent intertransaction k-itemsets [7].
In general, FITI consists of the following three phases.

1. Phase I: Mining and Storing Frequent IntraTransaction Itemsets
2. Phase II: Database Transformation
3. Phase III: Mining Frequent InterTransaction Itemsets

Phase I: Mining and Storing Frequent IntraTransaction Itemsets

In this phase, frequent intertransaction rules are mined and stored in data structure called as FILT (Frequent – Itemset Linked Table). Many fast algorithms are developed to mine intransaction rules which can be applied to this step as well. The space needed for FILT data structure is smaller as it only store Intransaction rules rather than Candidate itemsets [7].
FILT data structures

1. **Lookup Links:** The data structure consists of an itemset Hash Table, with nodes linked by several kinds of links. Each frequent intratransaction itemset is assigned a unique ID number that corresponds to a row number in the itemset Hash Table. Each itemset is stored in a node pointed to by a lookup link from the corresponding row in the table,

Example. Let {a},{b},{c},{e},{a,b},{a,c},{b,c},{a,b,c} be the frequent intratransaction itemsets derived by Apriori Each is then inserted into FILT by with the node pointed to by corresponding lookup link .Here ID of {a} is 1 and ID of {a,b,c} is 8

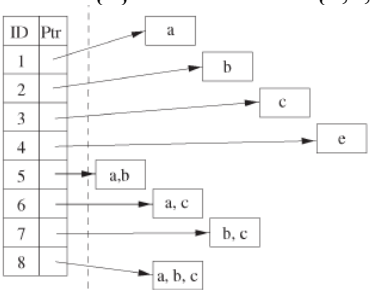

Figure 2. Lookup Links [1].

2**. Generator and Extension Links**: Given a node NF that contains an intratransaction k-itemset F, the generator links of NF point to the two (k-1)- itemsets that are combined to form F in the Apriori algorithm..The itemset {a,b,c} has its two generator links pointing to {a,b} and {a,c}.On the other hand {a,b} and {a,c} are combined to {a,b,c} both of them will have extension link pointing to {a,b,c}.Because of the nature of generator and extension links, they are depicted in the same diagram and the generator/extension relationship is represented by a bidirectional arrow.

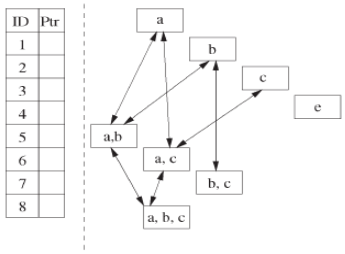

Figure 3. Generator and Extension Links [1].

3. **Subset Links**: Given a node NF that contains an intratransaction k-itemset F, the subset links of NF point to all subsets of F with size k - 1. For example, the subset links of {a, b, c} point to {a, b}, {a,c}and {b,c} in Fig.

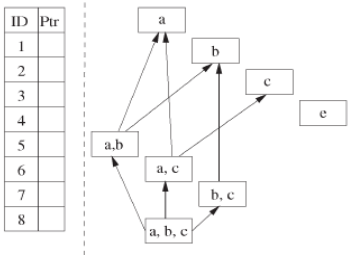

Figure 4. Subset Links [1].

4. **Descendant Links:** FILT is composed of an array and a hash-tree. Given a node NF that contains an intratransaction k-itemset F, the descendant links of NF point to all of its descendants in the hash-tree. If F = {e1…ek} then its descedents will be {e1,..ek, ek+1} For example, descedents of {a} will be {a,b} and {a,c}. Unlike the subset links, the descendant links of a node points to other nodes that share a common suffix with it.

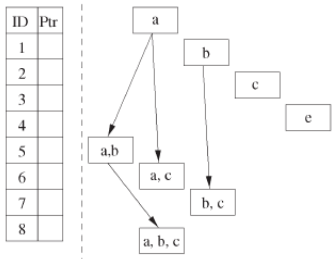

Figure 5. Descendant Links [1].

Phase II: Database Transformation

After forming the data structure FILT, the next step of FITI is to transform the database into a set of encoded Frequent-Itemset Tables, (called FIT tables) We have in total maxk FIT tables, {F1; . . . ; Fmaxk } where maxk is the

maximum size of the intratransaction itemsets discovered in Phase I. Each table Fk will be of the form, {di, IDseti} where di is the value of the dimensional attribute and IDseti is the IDs of frequent k-itemsets that are found in the transaction [7].

Phase III: Mining Frequent Intertransaction itemsets

After database transformation the next phase is to mine frequent intertransaction itemsets. Intertransactions itemsets are represented by their ID encoding form [2].

Algorithm for Mining of frequent intertransaction itemsets [2].

Input: A set of FIT tables: F1… Fmax, and the minimum
support threshold: minsup.
Output: The complete set of frequent intertransaction
itemsets
Generate frequent intertransaction 2-itemsets, L2; k=3;
While (Lk-1 ≠ Φ)
{
Generate candidate intertransaction k-itemsets, Ck;
Scan transformed database to update the count for Ck;
Let Lk ={c Є CK support(c) ≥ minsup};
k++;
}
Algorithm for Generation of all the (k-itemset) subsets of an intertransaction (k+1)-itemset [2].
Let S be the set of k-subsets of I;
S= {};
for(p=0; p < w; p++)
{
if (Ip! = 0)
{
If (Ip is an intratransaction one-itemsets) {
If (p!=0)
Add {I0,…, Ip-1, 0,…, Iw-1} to S
Else
Add {I0,…, Iw-1,0} to S
} else
{Let Ip be an intertransaction h-itemsets, h > 1
For each (h-1)-subset of Ip
{Let t be the ID of the (h-1)-subset
add {I0,…, Ip-1, t… Iw-1} to S}}}}
Return S;

3. Comparative study of EH-Apriori and FITI

EH-Apriori algorithm is based on Apriori approach to discover frequent intertransaction itemsets. Thus, both uses BFS (Breadth First Search) like level by level search so at each level database must be scanned, it generates large number of candidate patterns at each level, they are prone to memory shortage during mining process.

FITI algorithm provides faster performance than EH-Apriori. Currently the FITI algorithm is the state of the art in intertransaction association rule mining. However, the FITI introduces many unneeded combinations of items because the set of extended items is much larger than the set of items. The present form of FITI algorithm predicts along a single dimension it can be enhanced to n-dimensional intertransaction association-rules.

4. GROUP-BASED INTERTRANSACTION ASSOCIATION RULE MINING

Though FITI outperforms than EH-Apriori in efficiency, it still generates many unneeded combinations that should be avoided. Hence we have alternative approach of group based intertransaction association rule mining, where this group is set of transactions that meet a certain constraint.

This method of group-based inter-association mining can reduce the complexity of inter-transaction mining, as it also reduces the width of sliding windows and uses this set to replace extended itemsets, thus there is no need to consider too many combinations of extended items.

4. CONCLUSION AND FUTURE WORK

Association rule mining is the most tedious task in the intertransactions. In this paper we have reviewed EH-Apriori and FITI algorithms. These algorithms are compared according to their performance, we found that FITI is much better than EH-Apriori.FITI algorithm produces many extra and meaningless rules and makes the process complex. Thus we have stated another technique called group-based transactions to have efficient mining process.

In future, this method will be applicable to the Stock market data to predict and analyze the intertransaction rules and for knowledge discovery.

REFERENCES

[1]. Anthony K.H. Tung, Member, IEEE, Hongjun Lu, Member, IEEE, Jiawei Han, Member, IEEE, and Ling Feng, Member, IEEE.: Efficient Mining of Intertransaction Association Rules: IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 15, NO. 1, JANUARY/FEBRUARY 2003

[2]. Ping Li, Wenjing Xing, Huang Guangdong: Financial Asset Price Forecasting Based on Intertransaction Association Rules Mining: 2010 International Conference on E-Business and E-Government

[3]. Wanzhong Yang, Yuefeng Li, Yue Xu: Granule Based Intertransaction Association Rule Mining

[4] A. K.H. Tung, H.J. Lu, J.W. Han and L.Feng. Breaking the barrier of transactions: mining inter-transaction association rules, Proc. ACM SIGKDD Intl. Conf. Knowledge Discovery and Data Mining, 1999.

[5] Sotiris Kotsiantis, Dimitris Kanellopoulos: Association Rules Mining: A Recent Overview

[6] H. Lu, J. Han, L. Feng, Stock movement prediction and n-dimensional inter-transaction association rules, in: Proceedings of ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge, 1998, pp. 1–7.

[7] A. K.H. Tung, H.J. Lu, J.W. Han and L.Feng. Breaking the barrier of transactions: mining inter-transaction association rules, Proc. ACM SIGKDD Intl. Conf. Knowledge Discovery and Data Mining, 1999.