InterTARM: FP-tree based Framework for Mining Inter-transaction Association Rules from Stock Market Data

Hitesh Chhinkaniwala[1] P.Santhi Thilagam[2]
National Institute of Technology, Karnataka, India
[1]*hitesh@globalrenewableenergy500.com* [2]*santhi_soci@yahoo.co.in*

Abstract

Mining association rules from transactions occurred at different time series is a difficult task because of high computational complexity, very large database size and multidimensional attributes. Traditional techniques, such as fundamental and technical analysis can provide investors with tools for predicting stock prices. However, these techniques cannot discover all the possible relations between stocks and thus there is a need for a different approach that will provide a deeper kind of analysis. We propose a framework called InterTARM on real datasets. Our approach employs effective preprocessing, pruning techniques and available condensed data structure to efficiently discover inter-transaction association rules.

1. Introduction

Intra-transaction association rule mining is to find frequently appearing patterns for the items time series itself. Inter-transaction association rule mining is an extension of the intra-transaction association rule mining problem and includes the discovery of relationships, or itemsets, spanning transactions in one or more arbitrary dimensions [1] with defined inter-transaction interval. Comparing to classical association rules, complexity is more because the search space is much bigger as the number of possible rules increases dramatically with both the number of transactions and number of dimensions [2].

R_{11}: If the price of X and Y go up, 60% of time the price of Z goes up (Same day).

R_{12}: If the price of X and Y go up, 60% of time the price of Z goes up the next day.

R_{13}: If the price of X goes up more than 5% and traded volume goes down less than 50%, 60% of time the price of Z goes up less than 5% and traded volume goes up between 0 to 50% the next day.

It is obvious that investors/traders are more interested in the R_{13} kind of rules compare to R_{11} and R_{12}. R_{13} kind of rules is an important form of association rules, which is useful but could not be discovered with existing association rule mining framework. This kind of rule associates different itemsets among different transactions, along the axis of day.

2. Related Work

Since Agrawal et al. [3] introduced the concept of association rule mining, this problem has received a great deal of attention.

E-Apriori (Extended) and *EH-Apriori (Extended Hash)* [4] are first algorithms to mine inter-transaction association rules. *FITI (First Intra Than Inter)* [5] has been found much faster than *EH-Apriori* and its performance was found to be more acceptable in real life applications [6]. *E-Apriori*, *EH-Apriori* and *FITI* are *Apriori based algorithms* and inherits the drawback of scanning whole database which is very costly in case of mega-transactions. *EFP-tree (Extended Frequent Pattern Tree)* an *FP-tree based algorithm* in conjunction with *FP-growth* is so far best algorithm proposed by [2]. Experimental results show significant computational improvement of the *EFP-tree* over *FITI* when a large number of rules are present in data [2].

To summarize, mining inter-transaction association rules pose more challenges on efficient processing. However, the number of research papers on the inter-transaction mining problem is still few since it is a more challenging problem than intra-transaction mining and this area is still emerging.

3. Problem Statement

Let $\Sigma = \{e_1, e_2, ..., e_u\}$ be a set of u distinct items, W be a sliding window with w intervals, number of companies be n, number of attributes considered be a, user defined minimum support value *minsup* and

978-0-7695-3308-7/08 $25.00 © 2008 IEEE
DOI 10.1109/ICCSIT.2008.173

minimum confidence value *minconf*. Given the datasets of stock closing price *P* and traded volume *V*, our task is to mine the complete set of inter-transaction association rules of the form: X⇒Y, where, e_i represented as $x_i c_a Attr$ and e_j represented as $x_j c_b Attr$.

1. $X \subseteq \Sigma$
2. $\exists e_i \in X, 1 \leq i \leq u, e_i < w$
3. $\exists e_j \in Y, 1 \leq j \leq u, j \neq 0, e_j > e_i$
4. $X \cap Y = \Phi$
5. $1 \leq c_a, c_b \leq n$
6. $Attr \leq a$

4. InterTARM Framework

Although the association rule mining research has targeted in many directions, Luhr et al. [2] and Tung et al. [5] proposed a framework that can only discover inter-transaction association rules, whereas Srikant et al. [7] proposed an approach to mine quantitative intra-transaction association rules. In order to discover quantitative inter-transaction association rules from 1-dimensional transaction datasets of stock closing price and traded volume, a new approach called interTARM is developed.

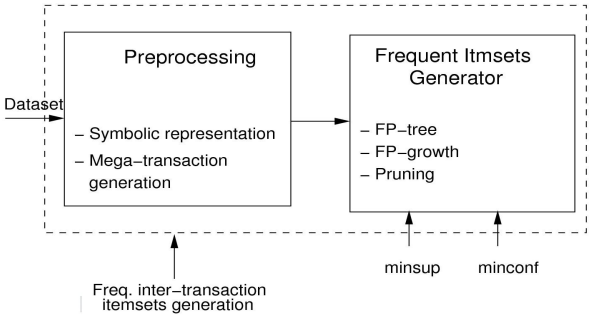

Figure 1. Process of mining inter-transaction association rules

InterTARM mines inter-transaction association rules for the dataset that contains constant number of items in each transaction. Proposed approach not only predicts the movement of stock price in either direction with user defined minimum support and confidence value but also predicts the probable variation in stock closing price and traded quantity based on historical data of attributes. InterTARM uses *FP-tree based algorithm* because it requires only two scan of transaction database to construct a tree and uses prefix-tree structure, which requires less memory [8]. For mining inter-transaction association rules, we adopted the steps shown in figure 1.

4.1. Preprocessing

Main objective of the preprocessing phase is to generate single mega-transaction dataset from two different datasets of closing price and traded volume. Table 1 and table 2 show example datasets of 10 trading days on closing price and traded volume respectively, where Cx denotes company x and Dy denotes day y.

Table 1. Example dataset - closing price

Day	C_1	C_2	C_3	C_4	C_5
D_1	305.3	148.4	158.9	114.1	314.0
D_2	302.3	144.9	153.0	116.3	320.2
D_3	295.9	140.5	150.0	117.6	328.6
D_4	294.0	141.1	149.6	115.7	332.5
D_5	291.7	142.0	150.7	117.0	330.8
D_6	294.7	141.0	151.9	116.3	332.1
D_7	296.4	139.0	152.4	113.0	330.6
D_8	289.5	132.6	155.3	116.1	339.3
D_9	299.3	135.7	155.4	120.9	344.0
D_{10}	297.7	138.4	155.5	124.9	345.4

Table 2. Example dataset - traded volume

(Values are in thousand)

Day	C_1	C_2	C_3	C_4	C_5
D_1	19.91	1012.83	69.53	27.73	1.65
D_2	12.81	491.85	160.40	114.21	4.90
D_3	14.16	354.95	165.36	122.13	9.71
D_4	16.14	481.66	89.51	43.67	56.81
D_5	19.46	642.45	176.90	74.74	3.16
D_6	10.50	403.98	224.03	52.68	4.13
D_7	12.37	408.74	196.52	19.45	7.66
D_8	15.75	1371.60	258.00	110.69	31.44
D_9	21.65	964.75	231.41	138.75	14.24
D_{10}	13.32	540.16	196.03	117.72	19.01

We provide symbolic representation to both input datasets separately and combine them as a single dataset. Sliding window is used to reduce the search space and generates mega-transactions, which fall within given sliding window size *w*.

4.1.1. Symbolic representation. Time series data are difficult to manipulate, but when they can be treated as symbols (item units) instead of data points, interesting patterns can be discovered and it becomes an easier task to mine. Change in closing price (C_P) and traded volume (C_V) in inter-day have been discretized and represented as single digit. We have grouped change in closing price (C_P) and traded volume (C_V) and combine them as described below.

Table 3. Symbolic representation of change in closing price and traded volume

(a)

C_1	C_2	C_3	C_4	C_5
3	3	4	2	2
3	4	3	2	1
3	2	3	3	2
3	2	2	2	3
2	3	2	3	2
2	3	2	4	3
3	4	2	1	1
1	2	2	1	2
3	2	2	1	2

(b)

C_1	C_2	C_3	C_4	C_5
4	4	1	1	1
2	3	2	2	2
2	2	4	4	1
2	2	2	2	2
4	4	2	3	2
2	2	3	4	2
2	1	2	1	1
2	3	3	2	4
4	4	3	3	2

The rationale behind grouping on assigned boundary values is that the companies belong to National Stock Exchange (NSE) 100 index are less volatile and owner's stack in these companies is more. In such cases change in closing price and traded volume are most of the times within set boundaries. Table 3 shows first level of symbolic representation.

$C_P > 2.5\%$ than 1, $0 \leq C_P \leq 2.5\%$ than 2,
$0 > C_P \geq 2.5\%$ than 3, $C_P < 2.5\%$ than 4,
$C_V > 50\%$ than 1, $0 \leq C_V \leq 50\%$ than 2,
$0 > C_V \geq 50\%$ than 3, $C_V < 50\%$ than 4

Table 4. Mapping table

11: 00	12: 01	13: 02	14: 03
21: 04	22: 05	23: 06	24: 07
31: 08	32: 09	33: 10	34: 11
41: 12	42: 13	43: 14	44: 15

Table 4 provides second level of symbolic notation by combining values of table 3a and table 3b. $D1_2$ in dimensional attribute suggests change occurred between day 1 to day 2. Now onwards Dx_y will be used as dimensional attribute.

4.1.2. Sliding window. Users may not be interested in the rules that span longer than a certain number of intervals. In order to avoid spending unnecessary resources to mine the rules which users are not interested in, a sliding window is introduced. Only those rules covered by the window is considered, which thus limits the number of possible rules. This is also reasonable in stock market prediction [4].

4.1.3. Mega-transaction. Mega-transaction or Extended-transaction [1][2] in a sliding window W is just the set of items in W, each appended with the corresponding subwindow number of the interval that contains the item.

Table 5. Modified symbols

Day	C_1	C_2	C_3	C_4	C_5
$D1_2$	11	11	12	04	04
$D2_3$	09	14	09	05	01
$D3_4$	09	05	11	11	04
$D4_5$	09	05	05	05	09
$D5_6$	07	11	05	10	05
$D6_7$	05	09	06	15	09
$D7_8$	09	12	05	00	00
$D8_9$	01	06	06	01	07
$D9_{10}$	11	07	06	02	05

Table 6. Mega-transactions retrieved

1111 1211 1312 1404 1504 2109 2214 2309 2405 2501 3109 3205 3311 3411 3504
1109 1214 1309 1405 1501 2109 2205 2311 2411 2504 3109 3205 3305 3405 3509
1109 1205 1311 1411 1504 2109 2205 2305 2405 2509 3107 3211 3305 3410 3505
1109 1205 1305 1405 1509 2107 2211 2305 2410 2505 3105 3209 3306 3415 3509
1107 1211 1305 1410 1505 2105 2209 2306 2415 2509 3109 3212 3305 3400 3500
1105 1209 1306 1415 1509 2109 2212 2305 2400 2500 3101 3206 3306 3401 3507
1109 1212 1305 1400 1500 2101 2206 2306 2401 2507 3111 3207 3306 3402 3505

Table 7. Frequent mega-transaction itemsets in order of their occurrences

2109 3109
1109 2109 3109 3305
1109 2109 3305
1109 1305 3306
1305 3109 3305
2109 3306
1109 1305 3306

To distinguish the items in a mega-transaction from the items in a traditional transaction, items in a mega-transaction are called as **extended-items**. Table 5 shows 1-dimensional transaction dataset with its dimensional attributes, trading day. In dataset, 9 such transactions are shown. Assuming that the length of sliding window *w* is 3, we will have 7 sliding windows. Table 6 shows generated extended-items. Notice that we do not count the last several transactions in a sequence, which do not contain the full span of the sliding window. Table 7 shows extended frequent items found in the first pass over the datasets. The frequent intra-items found given the minimum support threshold *minsup* = 3 are 1109:4, 1305:3 with frequency counts of 4 and 3 respectively. The frequent inter-items meeting the same *minsup* are 2109:4, 3109: 3, 3305: 3, 3306: 3. First digit in symbol gives

subwindow number contains the item, last two digits show attribute based on change in price as well volume and remaining digits provide company code.

4.2. Tree construction

We use an existing data structure called *FP-tree* [8]. To ensure that the tree structure is compact and informative, only frequent length-1 items should have nodes in the tree, and the tree nodes are arranged in such a way that more frequently occurring nodes will have better chances of node sharing than less frequently occurring ones [2].

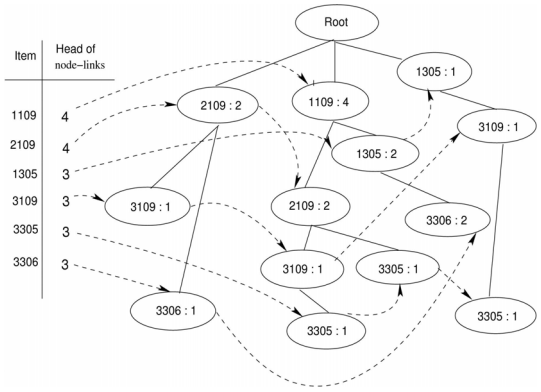

Figure 2. The FP-tree for the example dataset

Figure 2 shows the *FP-tree* for the frequent mega-transaction itemsets shown in table 7. To facilitate tree traversal, an item header table is built in which each item points to its first occurrence in the tree via a node-links. Sort frequent mega-transaction itemsets on their subwindow number after tree construction is over. If each sorted frequent itemsets has subwindow number '1' than store itemset in hash table else skip itemset because it is not satisfying inter-transaction criteria.

The *FP-growth* is currently one of the fastest approaches to frequent intra-transaction itemsets mining [9]. We adopt a pattern growth approach, which uses a divide and conquer approach that recursively builds the entire set of frequent associations by constructing trees conditioned on known frequent base itemsets and taking the dot products of the frequent items in the condition tree and conditional base itemsets [8]. The new conditional itemsets than become the conditional base for the next set of conditional trees. If all items in conditional base itemsets start with digit '1' than prune such itemsets.

5. Experimental Study

Dataset from Historical End-of-Day (EOD) Quotes for (NSE India) Stocks & Indices Version 7 [10] is used. The characteristics of the different datasets used for the experimental work are defined by several parameters as shown in table 8.

Table 8. Parameter settings

Parameter	SET_1	SET_2	SET_3
Filesize	1.5MB	1.05MB	650KB
Number of transactions	2400	1600	1000
Average length of intra-transactions	90	90	90
Interval span of inter-transactions	3	3	3

5.1. Performance analysis

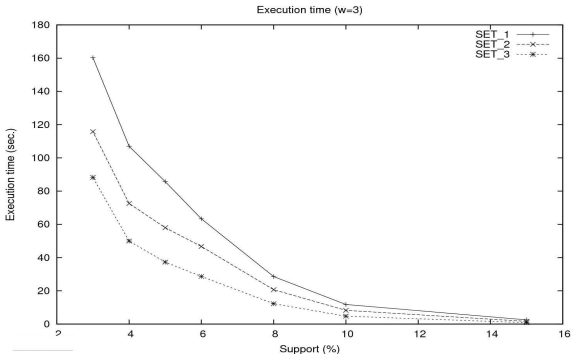

Figure 3. Execution time ($w = 3$)

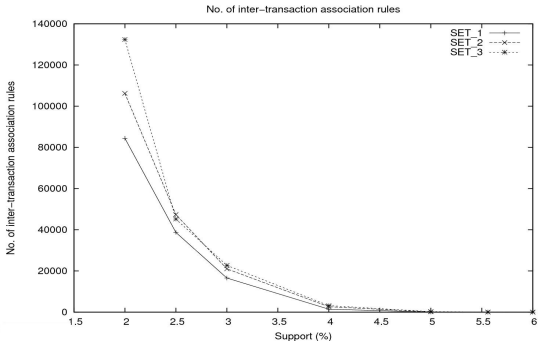

Figure 4. No. of inter-transaction rules

Figure 3 shows the execution time requires to mine inter-transactions association rules with InterTARM when support threshold is lowered from 15% to 2%. It concludes that an order of magnitude difference in the execution times exists due to large number of discovered rules at the lower support values as shown in figure 4. Figure 4 also admits the basic nature of inter-transaction rule mining i.e. number of rules increases exponentially once the minimum support barrier is broken.

Figure 5 shows the effect on execution time when number of transactions in database increases. As

number of transactions increases, large number of frequent itemsets are generated which requires more time for execution for given support value.

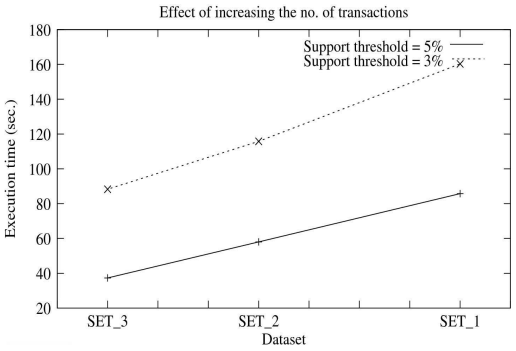

Figure 5. Effect of increasing the no. of transactions

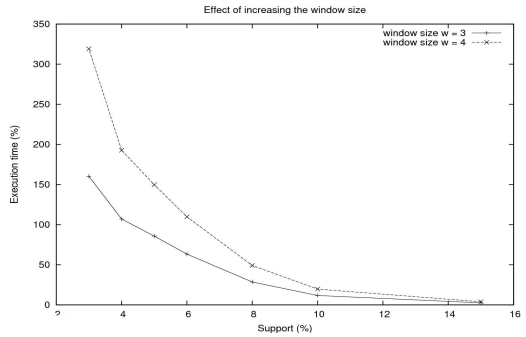

Figure 6. Effect of increasing the window size

To evaluate the effect of increasing sliding window size *w*, we vary *w* from 3 to 4. Number of items in mega-transactions is increasing as sliding *w* increases. The number of frequent inter-transaction itemsets increases with more items available in mega-transactions and which in turn requires more execution time. Figure 6 follows the same nature.

6. Conclusion

Proposed work has introduced a framework for the mining of association rules from highly fluctuating stock market environment and demonstrated their application through the discovery of inter-transaction association rules with user defined minimum support and confidence level. Experimental results show the normal phenomena of National Stock Exchange, India, as company stock price follows movement of index giants for reasonable higher support and confidence. Dataset used has equal probability of occurring right hand side event after the occurrence of left hand side event. We have considered 16 such right hand side associated events for each left hand side events and that gives equal probability of approximately 6.25% for each right hand side events to occur. Proposed work mine initial rules with 5.56% support and more than 30% confidence, which improves the level of confidence for investment and stock trading.

7. References

[1] S. Luhr, G. West, and S. Venkatesh, "Emergent intertransaction association rules for abnormality detection in intelligent environments", *In Proceedings of the Intelligent Sensors, Sensor Networks and Information Processing Conference*, December 2005, pp. 343-347.

[2] S. Luhr and S. Venkatesh, "An extended frequent pattern tree for inter-transaction association rule mining", *Technical Report*, 2005.

[3] R. Agrawal, T. Imielinski, and A. Swami, "Mining association rules between sets of items in large databases", *In Proceedings of the ACM SIGMOD International Conference on Management of Data*, May 1993, pp. 207-216.

[4] H. Lu, J. Han, and L. Feng, "Stock movement prediction and n-dimensional inter-transaction association rules", *In Proceedings of the ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery*, June 1998, pp. 1-7.

[5] A. K. H. Tung, H. Lu, J. Han, and L. Feng, "Breaking the barrier of transactions: Mining inter-transaction association rules". *Knowledge Discovery and Data Mining*, August 1999, pp. 297-301.

[6] C. Berberidis and L. Angelis, "Prevent: An algorithm for mining inter-transactional patterns for the prediction of rare events", *STAIRS'04: In Proceedings of the 2nd European Starting AI Researcher Symposium*, August 2004.

[7] R. Srikant and R. Agrawal, "Mining quantitative association rules in large relational tables", *In Proceedings of the 1996 ACM SIGMOD International Conference on Management of Data*, June 1996, pp. 1-12.

[8] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation", *In Proceedings of the ACM SIGMOD International Conference on Management of Data*, May 2000, pp. 1-12.

[9] L. Qin and Z. Shi, "Efficiently mining association rules from time series", *ICIC'05: Proceedings of International Conference on Intelligent Computing*, 2005.

[10] http://*www.idslindia.com*, "NSE India real time data for stocks & futures", *Intelligent Data Services*, January 2008.