International Journal of Computer Science and Applications, Vol. 5, No. 4, pp 57 - 69, 2008

PREDICTIVE MODELING OF INTER-TRANSACTION ASSOCIATION RULES – A BUSINESS PERSPECTIVE

D. BHANU [1], P. BALASUBRAMANIE [2]

[1] *MCA Department, Sri Krishna College of Engineering and Technology, Kuniamuthur,Coimbatore,TamilNadu, India*
bhanusaran@yahoo.com

[2] *Professor and Head, Department of Computer Science and Engineering, Kongu Engineering College, Perundurai, Erode, Tamilnadu, India*

Abstract

Traditional association rules are mostly mining intra-transaction associations i.e., associations among items within the same transaction where the idea behind the transaction could be the items bought by the same customer. In our work, we utilize the framework of inter-transaction association rules, which associate events across a window of transaction. The new association relationship breaks the barriers of transaction and can be used for prediction. Typically, the rules generated are derived from the patterns in a particular dataset. A major issue that needs more attention is the soundness of the rules outside of the dataset from which they are generated. When new phenomenon happens, the change in set of rules generated from the new dataset becomes more significant. In this paper, we provide a model for understanding how the differences between different situations affect the changes of the rules based on the concept of groups what we call factions. Also we provide a technique called Coalescent dataset, to generate set of rules for a new situation. Various experimental results are reported by comparing with real life and synthetic datasets, and we show the effectiveness of our work in generating rules and in finding acceptable set of rules under varying conditions.

Keywords: Association Rules; Intra-transaction associations; Inter-transaction associations; Factions; Coalescent Dataset.

1. Introduction

Armed with better information, the management of a company can apply their creativity and judgment to make better decisions and get better returns. The discovery of association rules from large databases has proven beneficial for companies since such rules can be very effective in revealing actionable knowledge that leads to strategic decisions. Association rule mining research [1] has progressed in various directions, including efficient, Apriori–like mining methods[2][16], mining generalized, multi-level or quantitative association rules [5][13][14], mining sequential patterns and episodes [9] [13],association rule mining query languages[10] and constraint based rule mining [7] [11] [7] .However, there is an important form of association rules which are useful but could not be discovered with the existing framework. A classical association rule expresses the associations among items within the same transaction. To also deal with intra and inter-transaction association analysis, we name the classical association rules as intra-transaction associations and the latter as inter-transaction associations. In view of this, we have utilized the formulation of a new type of rules called the inter-transaction association rules. We provide the definition of this new form of rules and used it for predicting new set of rules under different situations.

Much work in data mining revolves around the discovery of rules within large quantities of data. Rules over relations are of the form $C_1 \Rightarrow C_2$ where C_1 and C_2 are conditions on tuples of the relation. We utilize the framework of inter-transaction association rules, which associate events across a window of transaction. Research has been done on many aspects of Association Rule Mining Technique, but there is only a little focus on the applicability of the rules outside of the dataset from which these rules are derived. For a new occurrence, the decision can be made only with the rules generated from the dataset collected for an earlier situation. In this paper, we concentrate on extending the applicability of Association Rules. The model provides a framework for understanding the differences between the two sets of rules from datasets at different situations. The model is based on the concept of fine partitioned groups what we call factions. We provide a simple technique called the Coalescent Dataset, which acts as the source for generating new set of rules. The solution that we provide is applicable to retail stores. The remaining of the paper is organized as follows. The next section, Section 2, will define inter-transaction rules formally. Section 3 describes the problem and methodology in detail. Section 4 explains the experimental setup and the results and will conclude with Section 5.

2. Association Rule Mining

The notion of inter-transaction Association rule mining is introduced as follows:

Definition 2.1: Let ? = $\{e_1, e_2 \ldots . e_u\}$ be a set of literals, called items. Let D be an attribute and Dom (D) be the Domain of D. A transaction database is a database containing transactions in the form of (d, E), where $d \in$ Dom (D) and E is a $\subseteq$? . The attribute D in the transaction database is called the Dimensional attribute. It is assumed that the domain of the dimensional attribute is ordinal and can be divided into equal length intervals.

Definition 2.2: A sliding window W in a transaction database T is a block of w continuous intervals along domain D, starting from interval d_0 such that T contains a transaction at interval d_0. Each interval d_j in W is called a subwindow of W denoted as W[j], where $j = d_j - d_0$. We call j the subwindow number of d_j within W.

Definition 2.3: An intra-transaction itemset is a set of items $A \subseteq$? . An inter-transaction itemset is a set of extended items $B \subseteq \sum'$ such that $\exists\, e_i(0) \in B$, 1 = i = u.

Definition 2.4: An inter-transaction association rule is an implication of the form $X \Rightarrow Y$, where

1 $X \subseteq \sum'$, $Y \subseteq \sum'$

2 $\exists\, e_i(0) \in X$, 1 = i = u.

3 $\exists\, e_i(j) \in Y$, 1 = i = u, j ? 0,

4 X n Y = F

Similar to studies in mining intra-transaction association rules, we introduce two measures of inter-transaction association rules, support and confidence.

Definition 2.5: Let S be the number of transaction in the transaction database. Let T_{xy} be the set of transactions that contain a set of items $X \cup Y$ and T_x be the set of transactions that contain X. Then, the support and confidence of an inter-transaction association rule $X \Rightarrow Y$ are defined as

Support = $|T_{xy}| / S$

Confidence = $|T_{xy}| / |T_x|$

Given a minimum support level minsup and a minimum confidence level minconf, the task is to mine the complete set of inter-transaction association rules from the transaction database with support = minsup and confidence = minconf. The problem, of mining inter-transaction association rules can be decomposed into 2 sub problems:

1. Find all inter-transaction itemsets with support higher than minsup, which we call frequent itemsets.

2. For every frequent inter-transaction itemset F and for all possible combination of $X \subset F$, output a rule $X \Rightarrow (F-X)$, if its confidence is higher than minconf.

3. Related Work

Since the introduction of the problem of mining association rules [1], several generate-and-test type of algorithms have been proposed for the task of discovering frequent sets. An efficient breadth-first or level-wise method for generating candidate sets, i.e., potentially frequent sets has been proposed in [2] [4] [9]. This method also called Apriori is the core of all known algorithms except the original one [1] and its variation for SQL, which have been shown inferior to the level-wise method. [2][9][4]

An alternative strategy for a database pass, using inverted structures and a general purpose DBMS, has been considered. Other work related to association rules includes the problem of mining rules with generalization [3], management of large amount of discovered rules [6] and a theoretical analysis of an algorithm for a class of KDD problems including the discovery of frequent sets [9].A connectionist approach to mining rules is presented in [7]. All these approaches focus on generating rules, but a little attention is given to address the credibility of the rules outside of the dataset from which it was generated.

4. Problem Description

Importance of Locales

The importance of rules identified by the association rule mining is based on the validity and the credibility of the discovered rules. Consider a supermarket (company) managing a chain of stores that discovers association rules in the sale transactions at one of its stores. For opening a new store, the company would make use of the knowledge obtained from the discovered association rules. However, if the buying behaviour and people culture is different, the rules that are derived are not pertinent to the new store. From the strategic point of view, it is important from the point of decision maker to know which rules discovered from the transactions at the first store are applicable to the new store at a different location. Therefore, when applying association rules derived from one dataset to a new situation, it is necessary to ensure that the factors deriving the rules are consistent between the source of the rules and where the rules are applied. Hence, locales that determines the changes in the set of rules helps us understand the discrepancy in the rules observed at the two stores.

Problem Formulation

The problem focuses at estimating the association rules generated for a situation based on the data collected for another situation with a varied distribution of locales. The set of fields / attributes from which association rules are determined are called items. An attribute whose function is not primary, but whose value affects the relation between items is called a factor. Factors that possess demographic characteristics are used to identify a particular group of people having similar characteristics. Gender, age etc., can be considered as factors, which affect / change the buying behaviour in different situations. A set of factors with associated values that generates certain set of rules is called factions. We propose a method to generate factions. The model for generating association rules are as follows:

Every faction has a tight binding in generating set of association rules. The concept behind being, that people from the same faction will behave in the same way. Each situation has a mix of different factions in some proportions. The difference in the mix of proportions of the factions for different situations gives rise to the difference between the set of association rules for each situation.

The major task is to determine the association rules for a varying situation. The problem of estimating association rules can be done if we have the dataset for the first situation (the source dataset), the set of factions that make up the two situations, and the proportions of the factions for the new situation. Also, we define a method to construct Coalescent Dataset, which is a dataset sampled from the source dataset by the proportions of the factions for the new situation, to get a good estimate of the large itemsets for the new situation. Obtaining the dataset is not much difficult from which we can get source dataset with its corresponding background values. Construction of factions needs a method and is proposed. The proportions of factions vary and the survey approach is adopted to get the proportions of the new situation.

Coalescent Data Set

This is mainly based on the availability of sample representatives from each faction.

Coherent Sampling Technique

The behavior of faction is considered to be unswerving in any situation. Since the transactions of a particular faction contain all its patterns / rules for constructing a Coalescent Data set for a new situation, the faction with larger proportion in the overall population must have more transactions in the dataset to reflect its importance.

A fairly obvious way of reducing the database activity of knowledge discovery is to use only a random sample of the relation and to find approximate regularities. It is often important to know the frequencies and confidences of association rules exactly. In business applications, for example for large volumes of supermarket sales data, even very small differences can be significant. When relying on results from sampling alone, one also takes the risk of loosing valid association rules because their frequency in the sample is below the threshold.

Using a random sample to get approximate results is straightforward. Use a random sample to efficiently find a superset S of the collection of frequent sets. A superset can be determined efficiently by applying the level-wise method on the sample in the main memory, and by using a lowered frequency threshold. This approach requires one full pass over the database, and two passes in the worst case.

Frequent Set Discovery Using Sampling

We now apply the concept of negative border to find sampling for Frequent Sets [16]. It is not sufficient to locate a superset S of F(r, min_fr) using the sample and then to test S in r, because the negative border Bd^-(F(r, min_fr)) needs to be checked, too. If we have F(r, min_fr) subset of S, then obviously $S \cup Bd^-(S)$ is a sufficient collection to be checked. Determining $S \cup Bd^-(S)$ is easy: it consists of all sets that were candidates of the level-wise method in the sample. Algorithm 1 presents the principle: search for frequent sets in the sample, but lower the frequency threshold so much that it is very unlikely that any frequent sets are missed. Then evaluate the sets and their border, i.e., all sets that were evaluated in the sample, also in the rest of the database.

Algorithm 1

Input: A relation r over a binary schema R, a frequency threshold min_fr, sample size ss, and a lowered frequency threshold low_fr.

Output: The collection F(r, min_fr) of frequent sets and their frequencies, or its subset and failure report.

Method:

1. draw a random sample s of size ss from r;
2. compute S := F(s,low_fr) in main memory;
3. compute F := $\{ X \in S \cup Bd^-(S) \mid fr(X,r) = min\_fr \}$;
4. for all $X \in F$ do output X and fr(X,r);
5. report if there possibly was a failure;

In this case, Algorithm 1 reports that there possibly is a failure.

Therefore, now the problem is, given a database r and a frequency threshold min_fr, use a random sample s to determine a collection S of sets such that S contains with a high probability the collection of frequent sets F(r, min_fr).

Analysis of Sampling

As the next case, we analyze the relation of sample size to the accuracy of results. We first consider how accurate the frequencies computed from a random sample are. Samples of reasonable size provide good approximations for frequent sets.

Accuracy and Sample Size

We consider the absolute error of the estimated frequency. Given an attribute set X which is a subset of R and a random sample s from a relation over binary attributes R, the error e(X, s) is the difference of the frequencies:

$$e(X, s) = |\, fr(X) - fr(X, s)\, | \qquad (1)$$

where fr(x) is the frequency of X in the relation from which s was drawn.

To analyze the error, we consider sampling with replacement. The reason is that we want to avoid making other assumptions of the database size except that it is large. For sampling with replacement, the size of the database has no effect on the analysis, so the results apply in principle, on infinitely large databases. For very large databases there is practically no difference between sampling with and without replacement.

We analyze the number of rows in the sample s that contain X, denoted m(X, s). The random variable m(X, s) has binomial distribution, i.e., the probability of m(X, s) = c, denoted Pr [m(X, s) = c], is

$$P\,(|s|, c)\ fr\,(X)^{c}\,(1\text{-}fr(X))^{|s|\,-c} \qquad (2)$$

First, we consider the necessary size of the sample, given requirements on the size of the error. The following theorem gives a lower bound for the size of the sample, given an error bound e and a maximum probability d for an error that exceeds the bound.

Theorem 1: Given an attribute set X and a random sample s of size

$$|s| = 1/\,2\,e^{2}\ \ln 2/\,d \qquad (3)$$

The probability that e(X, s) > e is at most d.

Proof: We have:

$$Pr\,[e(X, s) > e] = Pr\{|\, fr(X, s) - fr(X)\,|\ |s| > e\ |s|\} \qquad (4)$$

The Chernoff bounds give an upper bound

$$2e^{-2(e\,|s|)\,2\,/\,|s|\ \sim}\ d \qquad (5)$$

for the probability.Table 1 gives values for the sufficient sample size |s|, for e = 0.01, 0.001 and d = 0.01, 0.001, 0.0001. With tolerable error, e around 0.01, samples of a reasonable size suffices. For many applications, these parameter values are perfectly reasonable – errors in the input data may be more likely than 0.0001. In such cases, approximate rules can be produced based on a sample. With tighter error requirements, the sample sizes can be quite large.

Table 1: Sufficient sample sizes, given e and d.

?	d	|s|
0.01	0.01	27,000
0.01	0.001	38,000
0.01	0.0001	50,000
0.001	0.01	2,700,000
0.001	0.001	3,800,000
0.001	0.0001	5,000,000

Construction of factions

The faction, which plays a major role in the formation of Coalescent data Set, should keep as much demographic information as possible, assuming that the people in each faction will have a similar behavior. For a faction, if it is a categorical or quantitative having only few values, it is meaningless to combine these values. However, if the factor is a quantitative attribute and the domain is large, we will need to partition the factors into intervals. A better method is to choose factions by commissioning a market research on the factors that determine the customer buying behavior for the product items that are going to be mined. The market research is acceptable and used by many people. Furthermore, an expert's opinion could be useful and lead to better factions. Another method is to partition the quantitative attributes according to the information entropy minimization heuristic that is widely used.

5. Experimental Results

We now explain the experiments we conducted in order to assess the practical feasibility of using samples of Coalescent Dataset for finding frequent sets and the set of factions generated for the datasets. We use synthetic datasets from departmental stores in our tests [2]. These databases model supermarket basket data . The central properties of the datasets are the following. There are ¦ R ¦ = 1000 attributes, and the average number T of attributes per row is 5, 10 and 20. The number ¦ r¦ of rows is approximately 100,000. The average size I of maximal frequent sets is 2, 4 and 6. Table 2 summarizes the parameters for the datasets. We assume that the real datasets from which association rules are discovered can be much larger than the test datasets. To make the experiments fair, we use sampling with replacement. We considered sample sizes from 20,000 to 80,000 since samples of these sizes are large enough to give good approximation. Table 2 illustrates the Dataset characteristics (T = row size on average, I = size of maximal frequent sets on average) and Table 3 illustrates the Lowered Frequency Thresholds (%) for d = 0.001

Table 2: Dataset Characteristics

Dataset name	$\lvert R \rvert$	T	I	$\lvert r \rvert$
T5.I2.D100K	1000	5	2	97,233
T10.I4.D100K	1000	10	4	98,827
T10.I6.D100K	1000	20	6	99,941

Table 3: Lowered Frequency Thresholds (%)

min_fr (%)	Sample Size			
	20,000	40,000	60,000	80,000
0.25	0.13	0.17	0.18	0.19
0.50	0.34	0.38	0.40	0.41
0.75	0.55	0.61	0.63	0.65
1.00	0.77	0.83	0.86	0.88
1.50	1.22	1.30	1.33	1.35
2.00	1.67	1.77	1.81	1.84

The lowered threshold depends on the frequency threshold and the sample size. The lowered threshold values are given in Table 3. We used in the computations the exact probabilities from binomial distribution.

Since the accuracy of the rules is directly dependent on the accuracy of the large itemsets, we demonstrate the efficiency of the technique from the quality of large itemsets generated. We tested our method with two different types of data: Synthetic data and real life data. In both cases, we use the data generation procedure described below to generate the datasets, one for the source and another for the target, respectively. We then generate the Coalescent dataset using the technique stated, and mine the Coalescent Dataset to obtain its large itemsets. The extent of matching between the source and the target dataset determines the accuracy of using only the source dataset, and the extent of matching between the coalescent dataset and the target determines the accuracy of the approach.

Data Generation

Synthetic Dataset: The popular method of generating transactions containing associations, presented by Agrawal and Srikant [2] is used. The same probability distributions are used and the same notations are taken [2]. We generate datasets for two situations: one is used as the source and the other as target. Each of the two datasets is partitioned into a common set of G factions. For each dataset, the generation program assigns random proportions to every faction. The relative proportion of a faction for the source has no relation to the relative proportion of any faction for the target. Given the total number of transactions in a dataset, the relative proportion of the faction determines the number transactions in that faction. The set of items involved in different factions is a major source that causes the difference of the large itemsets / rules among the factions. Therefore, the items assigned to factions are different from faction to faction. These two together give rise to the overall difference between the large itemsets of the two data sets.

Generating Itemsets: An itemset for a faction is generated by selecting initially the size of the itemset from a Poisson distribution with mean which is equal to |I|, the average size of an itemset. A fraction of items in an itemset is chosen from the previous itemset. An exponentially distributed random variable with mean that is equal to the correlation level decides this fraction for each itemset. This generates the fact that large itemsets usually have common items. The remaining items in the itemsets are picked according to the weights assigned with the items. The total number of itemsets for a faction is an input, to the program. There is a weight distribution n for the itemsets for each faction, which is derived from an exponential with unit mean. The weights correspond to the probability of choosing the corresponding itemsets when creating transactions.

Generating Transactions: Transactions are created by the addition of itemsets. All transactions for a particular faction are generated using the itemsets of that particular itemset. First, the size of the transaction is chosen from a Poisson distribution with mean given by an input parameter |T|. Itemsets are added to the transaction until the total number of item is equal to or exceeds the number of items for that transaction. When the length of the transaction is exceeded, the itemset is added to the transaction with a probability of 0.5. The Table 4 gives the various input parameters and the values used for them.

Table 4: Parameters for Data Generation

G	Number of Factions	25
|I|	Average size of itemset	4
|L|	Number of itemsets for a faction	500
|T|	Average size of transaction	10
N	Number of transactions in a data set	100000
C	Corruption level	Mean = 0.5 Variance = 0.1
	Average value for Correlation level	0.5
	Number of items in a faction	100
	Number of common items between factions	0,50,100

Real Life Data Sets:

The technique has been tested on the Census data set, which is downloaded from the UCI Repository of Machine learning databases [17] .For our purpose, we modify the dataset to form our database. We have used 14 attributes in our dataset. Even if the number of items is very large, each transaction has exactly 14 items. To create datasets the following method is adopted. Select some attributes as background attributes and set the remaining as foreground attributes. Generate the factions. Assign the factions for two data sets D1 and D2. Proportions of the factions for D1 are the same as that of the original database, but the size of the database is smaller than that of the original. Proportions of factions for D2 are different from the original database, and also with its size smaller than the original database. Generate datasets D1 and D2, according to their size and the corresponding proportions of factions.

Performance on Synthetic Data

For synthetic data, to detect any variation in the efficiency of the technique, the important aspect has been studied. All the factions involve the same set of items, half of the items for a faction is common to all and half is unique to that faction and all the items for a faction is distinct from the items in any other faction. For all the cases, the total number of items for each faction is set to 100. A support value of 5% is used as the threshold. Figure 1 shows the distribution of transactions with the specified threshold for the algorithm applied along time. Figure 2(a), 2(b), 2(c) shows the number of large itemsets predicted for same items, half same and half different items and distinct items respectively. Figure 2(d), 2(e) shows the distribution of error in support values for predicted itemsets for the same and distinct items.

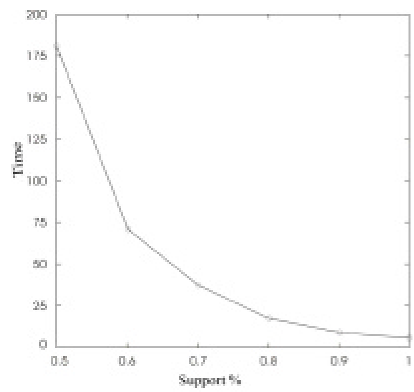

Figure – 1 – Distribution of transactions

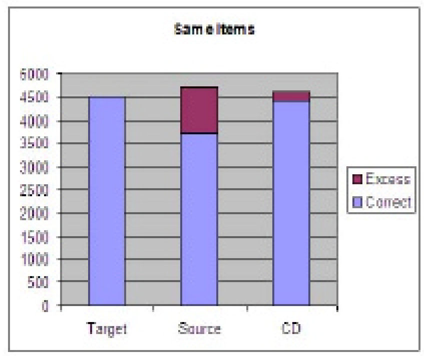

Figure – 2(a) – Number of itemsets predicted for same items

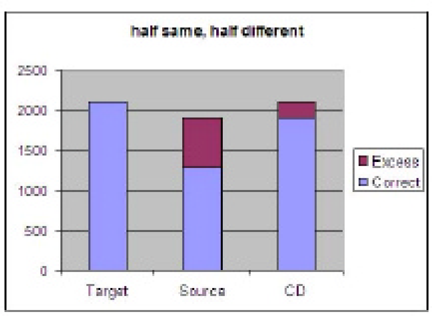

Figure 2(b) – Number of itemsets predicted for half same and half different Items

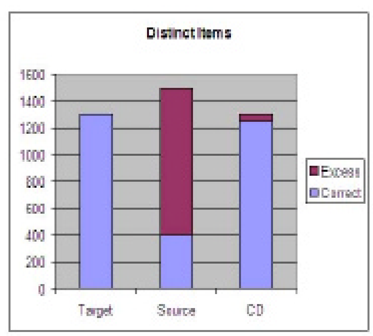

Figure 2(c) – Number of itemsets predicted for distinct items

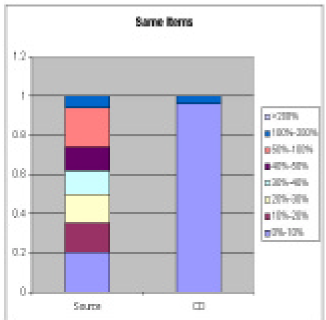

Figure – 2(d) – Distribution in error for same items

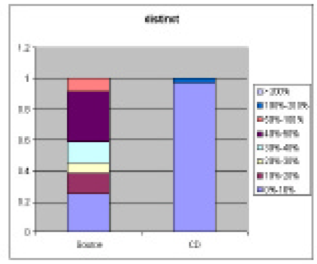

Figure - 2(e) – Distribution in error for for distinct items

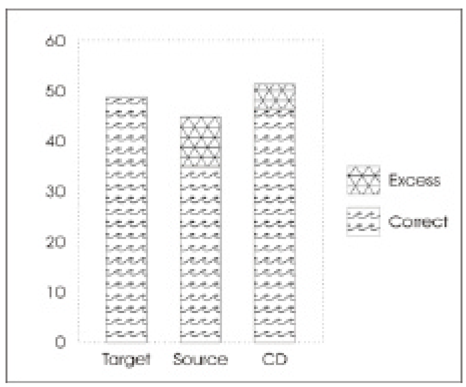

Figure – 3: Distribution of error with the direct and CD approach

In Figure 3, the bars marked Source and CD is associated with the direct approach and the Coalescent dataset approach. The target represents the distribution of predicated large itemsets. From the experiments conducted, we know that our technique is quite accurate in capturing the correct set of itemsets and also the values for their actual support.

6 Conclusions

The mining of association rules derived from data has been investigated in data mining. Most of the research is focused on optimizing the process, developing methods to handle different kinds of data, developing appropriate significance metrics, and improving the users control over the process. In this paper, we address the issue of extending the applicability of rules out of the existing datasets from where they were generated. We provide a model that distinguishes the difference between the situations using the concept of factions and Coalescent Dataset. Different situations have their own factions in different proportions. Using this model, we derive rules for a new situation, when given the datasets for the old situation.

References

[1] R. Agrawal, T. Imielinski, and A. Swami. Mining Association Rules between sets of items in large databases. In *Proc. of the 1993 ACM – SIGMOD Int. Conf. on Management of Data*, pages 207 – 216, Washington, D.C., USA, May 1993.

[2] R.Agrawal and R.Srikant. Fast algorithms for mining association rules. In *Proc. of the 1994 Int. Conf. on Very Large Data Bases*, pages 487 – 489, Santiago, Chile, September 1994.

[3] R.Agrawal and R.Srikant. Mining sequential patterns. In *Proc. of the 1995 Int. Conf. on Data Engineering*, pages 3 – 14, Taipei, Taiwan, March 1995.

[4] K. Ali, S. Manganaris, and R. Srikant. 'Partial Classification Using association rules", In *Proc. of the 3rd Int. Conf. on Knowledge Discovery in Databases and Data Mining,* Newport Beach, California, August 1997.

[5] M.Kamber, J.Han, and J.Y.Chiang. Metarule-guided mining of multi – dimensional association rule data cubes. In *Proc. of the 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97)*, pages 207 – 210, Newport Beach, California, August 1997.

[6] M.Klemettienen, .Mannila, P.Ronkainen, H.Toivonen, and A.I.Verkamo. Finding Interesting rules from large sets of discovered rules. In *Proc. of the 3rd Int.Conf. on Information and Knowledge Management,* pages 401 – 408, Gaithesburg, Maryland, November 1994.

[7] L.V.S. Lakshmanan, R.Ng, J.Han, and A.Pang. Optimization of constrained frequent set queries with 2-variable constraints. In *Proc. of the 1999 ACM – SIGMOD Int. Conf. on Management of Data*, Philadelphia, PA, June 1999.

[8] H.Lu, J.Han, and L.Feng. Stock movement and n-dimensional inter transaction association rules. In *Proc. of the 1998 SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD '98),* pages 12:1 – 12:7, Seattle, Washington, June 1998.

[9] H. Mannila, H. Toivonen, and A.I.Verkamo. Discovery of frequent episodes in event sequences. *Data Mining and Knowledge Discovery*, pages 1:259 – 289, 1997.

[10] R. Meo, G. Psaila, and S. Ceri. A new SQL-like operator for mining association rules. In *Proc. of the 1996 Int. Conf. on Very Large Data Bases*, pages 122-133, Bombay, India, September 1996.

[11] R.Ng, L.V.S. Lakshmanan, J. Han, and A. Pang. Exploratory mining and pruning optimizations of constrained association rules. In *Proc. of the 1998 ACM – SIGMOD Int. Conf. on Management of Data*, pages 13 – 24, Seattle, Washington, June 1998.

[12] J.S.Park, M.S.Chen, and P.S.Yu. An effective hash based algorithm for mining association rules. In *Proc. of the 1995 ACM – SIGMOD Int. Conf. on Management of Data*, pages 175 – 186, San Jose, CA, May 1995.

[13] R.Srikant and R. Agrawal. Mining generalized association rules. In *Proc. of the 1995 Int. Conf. on Very Large Data Bases*, pages 407 – 419, Zurich, Switzerland, September 1995.

[14] R. Srikant and R. Agrawal. Mining quantitative association rules in large relational tables. In *Proc. of the 1996 ACM – SIGMOD Int. Conf. on Management of Data*, pages 1 – 12, Montreal, Canada, June 1996.

[15] R.Srikant, Q. Vu, and R. Agrawal. Mining association rules with item constraints. In *Proc. of the 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97)*, pages 67 – 73, Newport Beach, California, August 1997.

[16] H.Toivonen. Sampling large databases for association rules. In *Proc. of the 1996 Int.Conf. on Very Large Data Bases*, pages 134 – 145, Bombay, India, September 1996.

[17] UCI Repository of Machine Leaning data bases. *Adult data set*, 1996.