Volume 4, Issue 2, February 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Experimental Survey on Data Mining Techniques for Association rule mining

Praveen Pappula, Ramesh Javvaji,
Assistant Professor,
SREC, Wgl, India,

Rama B
Assistant Professor,
Kakatiya University, Wgl, India

Abstract— In this paper, we give a survey on data mining techniques. More specially speaking, we talk about one important and basic data mining technique called association rule mining, which is to detect all subset of items which frequently occur and the relationship between them. This survey provides the related research results and also explored the future directions about data mining in Weather report, and it is a good reference for researcher on this topic.

Keywords— Data Mining, Association Rule Mining, Data Mining Techniques, Association Rule Mining for Weather Report

I. INTRODUCTION

The process of extracting useful patterns or information from large amount of data is known as data mining [1]. Most of the people think data mining as a synonym of knowledge discovery. But actually data mining can be considered as a step of knowledge discovery in databases (KDD). KDD process includes data cleaning (to remove noise and inconsistent data), data integration (where multiple data sources may be combined), data selection (where data relevant to the analysis task are retrieved from the database), data transformation (where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operations), data mining (an essential process where intelligent methods are applied in order to extract data patterns, pattern evaluation (to identify the truly interesting patterns representing knowledge based on some interestingness measures) and knowledge presentation (where visualization and knowledge representation techniques are used to present the mined knowledge to the user) [1] [7].

Data mining has attracted great deal of attention in information industry as well as in business areas because of the need of turning large data into useful information [4] [12]. Data mining is useful in an explanatory scenario in which there are no predefined notions about what will constitute an interesting outcome [9]. The database system industry has witnessed an evolutionary path in the development of the following functionalities like data collection and database creation, data management and advanced data analysis that includes data warehousing and data mining. Prediction and description are considered as two primary goals of data mining. Predictive data mining which produces the model of system described by the given data set and descriptive data mining, which produces new, non trivial information based on the available data set. The goal of prediction and description are achieved through data mining tasks such as classification, discovering association rules and clustering [11].

II. RELATED STUDY

Before discussing researches in specific areas of mining association rules, it is worth reviewing the theories behind association rules, the different types of rules, and their generation. Association rules are defined as statements of the form {X1,X2,…,Xn} -> Y [8], which means that Y may present in the transaction if X1,X2,…,Xn are all in the transaction. Notice the use of *may* to imply that the rule is only probable, not Identical. Note also, that there can be a set of items, not just a single item. The probability of Finding Y in a transaction with all X1, X2… Xn is Called confidence. The threshold (percentage) that a rule holds in all transactions is called Support. The level of confidence that a rule must exceed is called interestingness [15]. There are different types of association rules. The simplest form is the type that only shows valid or invalid association. This Boolean nature of the rule dubs the name Boolean Association Rules. In our market-basket example, "People who buy skim milk also buy low fat oil" is a Boolean association rule. Rules that aggregate several association rules, together are called Multilevel or Generalized Association Rules [15]. These rules usually involve a hierarchy and mining is done at a higher concept level. For example, "People who buy milk also buy bread". In this example, milk and bread each contains a hierarchy of different types and brands, but mining at the lowest level may not produce very interesting rules. A more complicated type of rules is the Quantitative Association Rules. This type of rules mines over quantitative (e.g. price) or categorical (e.g. gender) attributes, and is denoted in [1] by {<attribute:value>, <attribute:value>,…,<attribute:value>} -> <attribute:value>. For example, "People whose age is between 35 and 45 with income more than 250000 per year buy cars over 30000". However, the above types do not address the fact that transactions are temporal in nature. For example, mining before a product is introduced to or after a product is discontinued from the market will both adversely affect the support threshold. In view of this, [17] introduced the concept of an attribute's lifetime into the mining algorithm of Temporal Association Rules. In spite of the various kinds of rules, the algorithm to discover association rules can generally be broken down into two steps:

(1) Find all large (frequent) itemsets -A large item set is a set of items that exceeds the minimum support.
(2) Generate rules from the large item sets.

Since its introduction in [18], the Apriori algorithm has been the most mentioned algorithm for step 1. Many improvement [7, 13], e.g. speed up and scale up, of step 1 are about improving the Apriori algorithm by addressing its fallacy of generating too many candidate item sets. There are also algorithms that are not based on Apriori [1,10, 13] but aim at addressing the issues of speed of Apriori. Step 2 is mostly characterized by confidence and interestingness. There are researches about different way of generating rules [8] and alternative measure to interestingness [16]. There are also researches about generating different types of rules [1, 17].

III. DATA MINING TECHNIQUES

Various algorithms and data mining techniques like Classification, Clustering, and Association Rules etc are used for knowledge discovery [1] [6]. We have described the following data mining techniques.

A. Association Rules

Association rule mining tries to find frequent item set among large data sets [3]. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories and describes association relationship among different attributes. Such finding helps businesses to make certain decisions like customer's behavior analysis. However the number of possible Association Rules for a given dataset is generally very large and most of them are usually of less value. Association mining is an important research area in data mining.

B. Clustering

Clustering is one of the well known data mining technique and can be defined as the identification of similar classes of objects [5]. It is a common descriptive task in which one seeks to identify a finite set of categories or clusters. By using clustering techniques we can discover the overall distribution pattern and correlations among data attributes. Classification approach can also be used for effective means of distinguishing groups or classes.

C. Classification

Classification is the most commonly applied data mining technique, which employs a set of pre-classified examples to develop a model that can classify the population of records at large [2]. It is the discovery of a predictive learning function that classifies a data item into one of the several predefined classes. Fraud detection and credit risk applications are particularly well suited to this type of analysis. The data classification process involves learning and classification. In Learning the training data are analysed by classification algorithm.

IV. ASSOCIATION RULES

One of the well techniques of data mining is association rules which are used to find out the relationship or association between various items. The problem of finding relation between items is often termed as market basket analysis. In this problem the presence of items within baskets is identified so that the customers buying habits can be analysed. The technique is used in inventory management, sales promotion etc [11]. The discovery of association rules is primarily dependent on finding the frequent sets. This can require multiple passes through the database. The algorithms aims at reducing number of passes by generating a candidate set which should turn out to be frequent sets. Many different algorithms are designed to find out the association rules. The algorithm differs on the basis of how they handle candidate sets and how they reduce number of scans on the database. Some of the recent algorithms of association rule mining do not create candidate set. Practically the frequent sets generated are very large in number and this can be constrained by selecting only those items in which the user is interested. Let us consider a set of items and a transaction database which is again a set of transactions. The association rule takes the following form for a transaction database: X=>Y, where X and Y are the sets of items called item sets. There are two important basic measures for association rules, *support*(*s*) and *Confidence(c*). Since the database is large and users concern about only those
Frequently purchased items, usually thresholds of support and confidence are pre- defined by users to drop those rules that are not so interesting or useful. The two thresholds are called *minimal support* and minimal confidence respectively, additional constraints of interesting rules also can be specified by the users. The two basic parameters of Association Rule Mining (ARM) are: support and confidence.

Support(*s*) of an association rule is defined as the percentage/fraction of records that contain X *[Y* to the total number of records in the database. The count for each item is increased by one every time the item is encountered in different transaction T in database D during the scanning process. It means the support count does not take the quantity of the item into account. For example in a transaction a customer buys three bottles of beers but we only increase the support count number of *f*beer*g* by one, in another word if a transaction contains a item then the support
count of this item is increased by one. Support(*s*) is calculated by the following
formula:

$$Support(XY) = \frac{\text{Support count of } XY}{\text{Total number of transaction in } D}$$

From the definition we can see, support of an item is a statistical significance of an association rule. Suppose the support of an item is 0.1%, it means only 0.1 percent of the transaction contain purchasing of this item. The retailer will not pay much attention to such kind of items that are not bought so frequently, obviously a high support is desired for more interesting association rules. Before the mining process, users can specify the minimum support as a threshold, which means they are only interested in certain association rules that are generated from those item sets whose supports exceed that threshold. However, sometimes even the items sets are not as frequent as defined by the threshold, the association rules generated from them are still important. For example in the supermarket some items are very expensive, consequently they are not purchased so often as the threshold required, but association rules between those expensive items are as important as other frequently bought items to the retailer.

Confidence of an association rule is defined as the percentage/fraction of the number of transactions that contain X*[*Y to the total number of records that contain X, where if the percentage exceeds the threshold of confidence an interesting association rule X,Y can be generated.

$$Confidence(X,Y) = \frac{Support(XY)}{Support(X)}$$

Confidence is a measure of strength of the association rules, suppose the confidence of the association rule X)Y is 80%, it means that 80% of the transactions that contain X also contain Y together, similarly to ensure the interestingness of the rules specified minimum confidence is also predefined by users.

Association rule mining is to find out association rules that satisfy the predefined minimum support and confidence from a given database [Agrawal and Srikant 1994]. The problem is usually decomposed into two subproblems. One is to find those itemsets whose occurrences exceed a predefined threshold in the database; those item sets are called frequent or large itemsets. The second problem is to generate association rules from those large itemsets with the constraints of minimal confidence. Suppose one of the large itemsets is *Lk*, *Lk= fI*1, *I*2, ¢ ¢ ¢, *Ik¡*1,*Ikg*, association rules with this itemsets are generated in the following way: the first rule is *fI*1, *I*2, ¢ ¢ ¢, *Ik¡*1*g) fIkg*, by checking the confidence this rule can be determined as interesting or not. Then other rule 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. Those processes iterated until the antecedent becomes empty. Since the second sub problem is quite straight forward, most of the researches focus on the ¯rst sub problem.

V. METHODS TO DISCOVER ASSOCIATION RULES FOR WEATHER

The association rule mining is the method that finds out the association rules which satisfy the predefined minimum support and minimum confidence. The association rule mining is usually carried out in 2 steps. In the first step those items from the database are found out which exceed the predefined threshold. Such items are stated as frequent items or big items. In the second step the association rules are generated out of frequent items found in first step.

Various algorithms like Apriori algorithm, *Generalized Sequential Pattern* algorithm, partition algorithm, pincer search algorithm, dynamic item set counting algorithm, FP tree growth algorithm, Éclat and dEclat etc have been developed to find out the frequent items from the transaction database [1].

Apriori Algorithm. The Apriori algorithm is the most general and widely used association rule mining algorithm [10]. It uses an iterative method called layer search to generate (k+1) item sets from k item sets. The concept of Apriori and Apriori Tid was given by 1994 Agrawal et al. Other algorithms like SETM and AIS were also used for association rule mining but the performance of Apriori and Apriori Tid was better than these algorithms. This is because SETM and AIS generated too many candidate sets which were later found out to be infrequent among data sets. With large amount of data and with the advent of parallel computing technology various association mining algorithms like count distribution algorithm, data distribution algorithm, candidate distribution algorithm and improved Apriori algorithms have been proposed [3] [10]. These algorithms can be used under cloud computing environment.

Reducing time for generating frequent item sets can boost the performance of an association rule mining algorithm. Keeping this in mind various other algorithms were developed later. The concept of hashing can be used for pruning (removing infrequent item sets) which reduces time to generate frequent item sets. The process of association rule mining can be fastened by removing the infrequent item sets as quickly as possible though pruning can be problematic sometimes. The frequent patterns algorithm without candidate generation eliminates the costly candidate generation. It also avoids scanning the database again and again. So, we can use Frequent Pattern (FP) Growth ARM algorithm that is more efficient structure to mine patterns when database grows. FP tree growth algorithm is also used for mining and it does not create the candidate set. It rather creates a tree like structure to find the frequent sets [13].

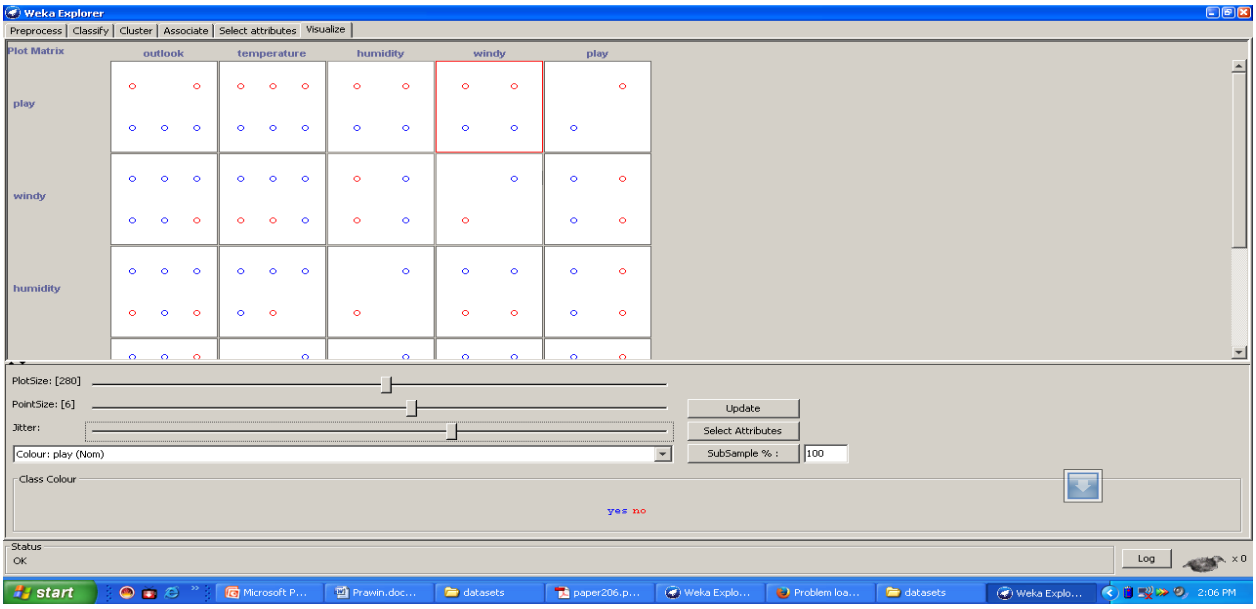
Input:
Database D
Mini Support ²
Mini Confidence »
Output:
Rt All association rules

Method:

01 *L*1 = large 1-itemsets;
02 for (k=2; *Lk¡*1 *6*= Â; k++) *do begin*
03 *Ck* =apriori-gen (*Lk¡*1); *//generate new candidates from Lk¡1*
04 for all transactions T *2* D *do begin*
05 *Ct*=subset (*Ck*, T); *//candidates contained in T.*
06 for all candidates C *2 Ct* do
07 Count(C) =Count(C) +1; *// increase support count of C by 1*
08 end
09 *Lk=f*C *2 Ct j* Count(C) ¸ ² £ *j* D *j g*
10 end
11 *Lf*=S*k Lk*;
12 *Rt* =Generate Rules (*Lf, »)*

Fig.1. Apriori Algorithm.

=== Run information ===
Scheme: Weka.Associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1
Relation: Weather.symbolic
Instances: 14
Attributes: 5 {outlook, temperature, humidity, windy, play }
=== Association model (full training set) ===
Apriori
=======
Minimum support: 0.15 (2 instances)
Minimum metric <confidence>: 0.9
Number of cycles performed: 17
Generated sets of large itemsets:
Size of set of large itemsets L (1): 12
Size of set of large itemsets L (2): 47
Size of set of large itemsets L (3): 39
Size of set of large itemsets L (4): 6
Best rules found:
1. outlook =overcast 4 ==> play=yes 4 conf:(1)
2. temperature=cool 4 ==> humidity=normal 4 conf :(1)
3. humidity=normal windy=FALSE 4 ==> play=yes 4 conf:(1)
4. outlook=sunny play=no 3 ==> humidity=high 3 conf :(1)
5. outlook=sunny humidity=high 3 ==> play=no 3 conf:(1)
6. outlook=rainy play=yes 3 ==> windy=FALSE 3 conf:(1)
7. outlook=rainy windy=FALSE 3 ==> play=yes 3 conf:(1)
8. temperature=cool play=yes 3 ==> humidity=normal 3 conf:(1)
9. outlook=sunny temperature=hot 2 ==> humidity=high 2 conf :(1)
10. temperature=hot play=no 2 ==> outlook=sunny 2 conf:(1)



GSP Algorithm (*Generalized Sequential Pattern* algorithm) is an algorithm used for sequence mining. The algorithms for solving sequence mining problems are mostly based on the a priori (level-wise) algorithm. One way to use the level-wise paradigm is to first discover all the frequent items in a level-wise fashion. It simply means counting the occurrences of all singleton elements in the database. Then, the transaction is filtered by removing the non-frequent items. At the end of this step, each transaction consists of only the frequent elements it originally contained. This modified database becomes an input to the GSP algorithm. This process requires one pass over the whole database

GSP Algorithm makes multiple database passes. In the first pass, all single items (1-sequences) are counted. From the frequent items, a set of candidate 2-sequences are formed, and another pass is made to identify their frequency. The frequent 2-sequences are used to generate the candidate 3-sequences, and this process is repeated until no more frequent sequences are found. There are two main steps in the algorithm.

1. Candidate Generation. Given the set of frequent (k-1)-frequent sequences F(k-1), the candidates for the next pass are generated by joining F(k-1) with itself. A pruning phase eliminates any sequence, at least one of whose subsequences is not frequent.
2. Support Counting. Normally, a hash tree–based search is employed for efficient support counting. Finally non-maximal frequent sequences are removed.

F1 = the set of frequent 1-sequence k=2,
do while F(k-1)!= Null;
Generate candidate sets Ck (set of candidate k-sequences);
For all input sequences s in the database D
do
Increment count of all a in Ck if s supports a
Fk = {a Є Ck such that its frequency exceeds the threshold}
k= k+1;
Result = Set of all frequent sequences is the union of all Fks
End do
End do

Fig 2.GSP Algorithm

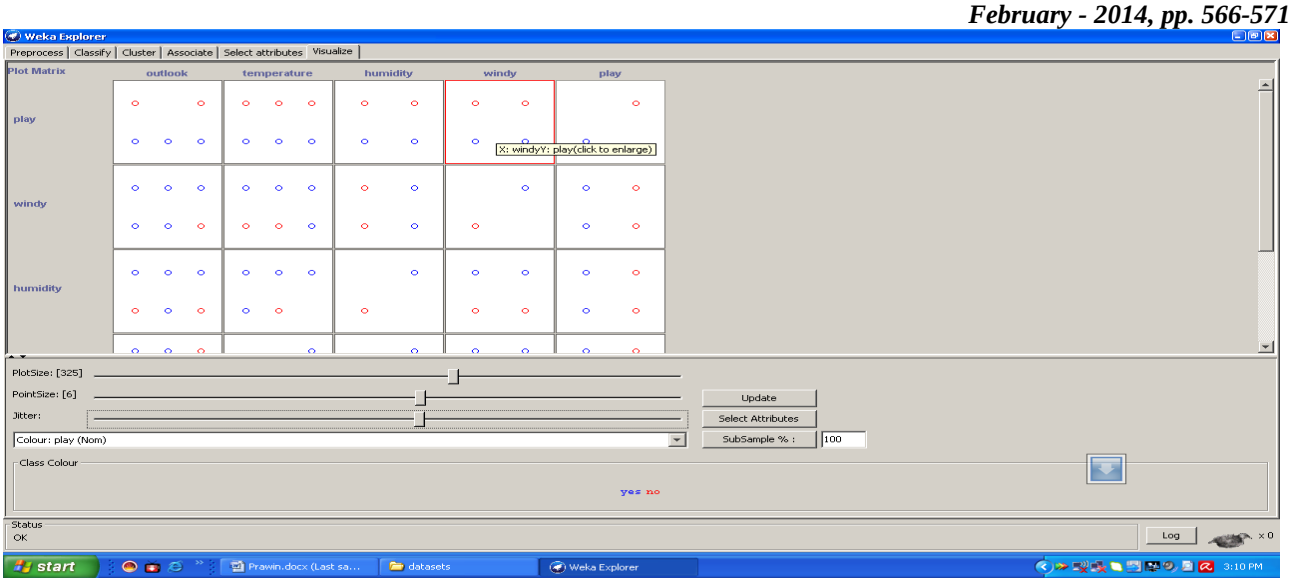
The above algorithm looks like the, Apriori algorithm one main difference is however the generation of candidate sets. Let us assume that: A → B and A → C are two frequent 2-sequences. The items involved in these sequences are (A, B) and (A, C) respectively. The candidate generation in a usual Apriori style would give (A, B, C) as a 3-itemset, but in the present context we get the following 3-sequences as a result of joining the above 2- sequences
A → B → C, A → C → B and A → BC

The candidate–generation phase takes this into account. The GSP algorithm discovers frequent sequences, allowing for time constraints such as maximum gap and minimum gap among the sequence elements. Moreover, it supports the notion of a sliding window, i.e., of a time interval within which items are observed as belonging to the same event, even if they originate from different events.

```
=== Run information ===
Scheme:       weka.associations.GeneralizedSequentialPatterns -S 0.9 -I 0 -F -1
Relation:     weather.symbolic
Instances:    14
Attributes:   5

              outlook
              temperature
              humidity
              windy
              play
=== Association model (full training set) ===

GeneralizedSequentialPatterns
============================
Number of cycles performed: 2
Total number of frequent sequences: 3
Frequent Sequences Details (filtered):
- 1-sequences
[1] <{high}> (3)
[2] <{FALSE}> (3)
- 2-sequences
[1] <{high,FALSE}> (3)
```

V. CONCLUSION

In this paper, we surveyed the list of existing association rule mining techniques. The topic of discovering association rules has been studied over couple of decades. Most of the foundation researches have been done. A lot of attention was focus on the performance and scalability of the algorithms, but not enough attention was given to the quality (interestingness) of the rules generated. In the coming decades, the trend will be to turn the attention to the application of these researches in various areas of our lives, e.g. genetic research, medicine, homeland security, and Weather forecasting etc.

REFERENCES

[1] Jiawei Han Micheline Kamber, *Data Mining concepts and techniques*, 2nd Edition.
[2] Jing Ding, Shanlin Yang, ―*Classification Rules Mining Model with Genetic Algorithm in Cloud Computing*‖, International Journal of Computer Applications (0975 – 888), Volume 48– No.18, June 2012.
[3] Ling Juan Li, min Zhang, ―*The strategy of mining association rule based on cloud computing*‖, International conference on business computing and global information, 2011.
[4] Ven Katadri .M, Dr. Lokaanaathaa C.Reddy, ―*A review on data mining from past to future*‖, International Journal of Computer Applications (0975 – 8887) Volume 15– No.7, February 2011.
[5] A. Mahendiran, N. Saravanan, N. Venkata Subramanian and N. Sairam, ―*Implementation of K-Means Clustering in Cloud Computing Environment*,‖ Research Journal of Applied Sciences, Engineering and Technology 4(10): 1391-1394, 2012.
[6] Musa J. Jafar, ―*A Tools-Based Approach to Teaching Data Mining Methods*‖, Journal of Information Technology Education, Volume 9, 2010.
[7] *http://en.wikipedia.org/wiki/Data_mining/*
[8] Data Mining Lecture Notes - *http://wwwdb.stanford.edu/~ullman/mining/mining.html.*
[9] Joanna Gordon, Chiemi Hayashi, ―*Exploring the Future of Cloud Computing*‖ World Economic Forum, 2010.
[10] Lu, Lin Pan, Rongsheng Xu, Wenbao Jiang,‖ *An Improved Apriori-based Algorithm for Association Rules Mining*‖ Sixth International Conference on Fuzzy Systems and Knowledge Discovery, 2009. FSKD '09.
[11] Arun K Pujari, *Data Mining Techniques*, second edition.
[12] Margaret H. Dunham, Data mining: *Introductory and Advanced Topics, Eighth impression*.
[13] Sanjeev Rao, Priyanka Gupta, ―*Implementing Improved Algorithm Over APRIORI Data Mining Association Rule Algorithm*‖, ISSN: 0976-8491 (Online) | ISSN: 2229-4333 (Print) IJCST Vol. 3, Issue 1, Jan. - March 2012.
[14] Rajesh Natarajan, B. Shekar: *Data mining (DM): poster papers: Relatedness-based data driven Approach to determination of Interestingness of association rules. Proceedings of the 2005 ACM symposium on applied computing SAC '05.* March 2005.
[15] Ming-Syan Chen, Jiawei Han, Philip S. Yu: Data Mining: *An Overview from a Database Perspective*. *IEEE Trans. On Knowledge And Data Engineering*. 1996.
[16] Keith C. C. Chan, Wai-Ho Au: *An effective algorithm for mining interesting quantitative association rules*. Proceedings of the 1997 ACM Symposium on Applied computing. April 1997.
[17] Juan M. Ale, Gustavo H. Rossi: *An approach to discovering temporal association rules*. Proceedings of the 2000 ACM symposium on Applied computing – *Volume 1*. March 2000.

