INTERACTIVE DATAMINING PROCESS BASED ON HUMAN-CENTERED SYSTEM FOR BANKING MARKETING APPLICATIONS

Olivier Couturier, Engelbert Mephu Nguifo
CRIL CNRS FRE 2499 - Université d'artois
rue Jean Souvraz, SP 18, F-62307 Lens Cedex, France

Brigitte Noiret
Caisse d'Epargne du Pas de Calais (CEPDC)
1, place de la république, B.P. 199, F-62304 Lens Cedex, France

Keywords: Human-Computer Interaction, Knowledge Discovery in Databases (KDD), Association Rules Mining, Banking Marketing.

Abstract: Knowledge Discovery in Databases (KDD) is the new hope for marketing due to the increasing collection of large databases. There is a paradox because the companies must improve the development policy of customer loyalty by using methods that do not allow to treat large quantities of data. Our current work is the results of a study that we led on a association rules mining in banking marketing problem. Our first encouraging results steered our work towards a hierarchical association rules mining, using a user-driven approach rather than an automatic approach. The user is at the heart of the process, playing a role of evolutionary heuristic. Mining process is oriented according to intermediate expert's choices. The final aim of our approach is to use the advantages of the methods to decrease both number of generated rules and expertise time. This paper presents the results of our research step for including the user into datamining process.

1 INTRODUCTION

Marketing process is a social and financial mechanism in which people satisfy their needs and desires by creating and exchanging products or other entities. The purpose is to establish satisfactory relations between customers and suppliers in order to preserve the trade preference. The final aim is to go towards a network which constitutes the effective capital of the companies. The sale of new products or maintenance products is a necessary step in order to secure customer loyalty. The problem is to anticipate their choices so as to predict the products which they might find interesting. There are two categories of customers. First of all, the customers whom the company wishes to keep as a customer and secondly, those from the other banks. For the first case, the term which is usually used is *defensive marketing* while for the second case, the corresponding term is *offensive marketing*. Our work is oriented towards the first problem.

Let P, be the population and S, be a sub-population. The problem is to find the better sub-population S in P depending on the initial problem. The implementation of a campaign on launching a new product can be costly depending on the number of people who must be contacted. The main idea is to have a maximum of positive answers in order to reduce expenses. It is necessary to be able to offer the products as well as possible by using the database history. Indeed, databases contain a significant quantity of knowledge which is hidden in meaningful masses. The association rules mining is one of the possible solutions allowing to solve the problem by establishing logical relations between products. The next section describes association rules mining.

As the number of large databases increasing, extracting useful information is a difficult and opened problem. This is the goal of an active research domain : Knowledge Discovery in Databases (KDD) (Fayyad et al., 1996). We focus on association rules mining (Agrawal et al., 1996) such as "If Antecedent then Conclusion". Several works were published which are based on two main indices to measure the quality of a rule: support and confidence (Agrawal et al., 1996). Two main limitations occurs from these works.

Couturier O., Mephu Nguifo E. and Noiret B. (2005).
INTERACTIVE DATAMINING PROCESS BASED ON HUMAN-CENTERED SYSTEM FOR BANKING MARKETING APPLICATIONS.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems*, pages 104-109
DOI: 10.5220/0002550301040109

First, the support does not allow to extract specific information valid on a small number of transactions in the database. Trivial information may already be known by the user. The calculation of this information has a high time cost. This type of approach is an automatic approach.

Secondly, the number of generated rules is too significant and leads to another problem called "Knowledge Mining" (Blanchard et al., 2003). Expertise time is costly and it is not taken into account in studies. Indeed, if the support decreases, the number of generated rules increases. The solution consists in using quality measures to rate the rules but the experience of the expert is not taken into account either. In order to exploit his tacit knowledge, hierarchical association rules mining can be used, in order to generate knowledge with various levels of granularity (Han and Fu, 1995; Hipp et al., 1998; Srikant and Agrawal, 1997; Srikant et al., 1997; Tseng, 2001).

In a real-expert datamining process, it is not always possible to formalize the tacit knowledge of an expert in order to optimize the rules because he does not always know which kind of rules he would like to obtain. To solve it, the expert is introduced while the process is ongoing. Our approach deals with this case. It is called a user-driven approach and it gives the main role of evolutionary heuristic to the user (Kuntz et al., 2000). The main idea of our approach is to use the advantages of the two methods to decrease both expertise time and number of generated rules. The hierarchical association rules mining is used to decrease the number of generated rules and the user-driven approach is used to reduce trivial rules.

Section 2 recalls association rules mining and gives a brief overview of both automatic and user-driven approaches. Section 3 describes our hybrid approach, and experimental results are presented in section 4.

2 ASSOCIATION RULES MINING

Association rules mining (Agrawal et al., 1996) can be divided into two subproblems: the generation of the frequent itemsets lattice and the generation of association rules. The complexity of the first subproblem is exponential. Let $|\mathcal{I}| = m$ the number of items, the search space to enumerate all possible frequent itemsets is equal to 2^m, and so exponential in *m* (Agrawal et al., 1993).

2.1 Problem

Let $\mathcal{I} = \{a_1, a_2, ..., a_m\}$ be a set of items, and let $\mathcal{T} = \{t_1, t_2, ..., t_n\}$ be a set of transactions establishing the database, where every transaction t_i is composed of a subset $X \subseteq \mathcal{I}$ of items where each transaction has a unique identifier, called TID. A set of items $X \subseteq I$ is called *itemset*. A subset of items $X \subseteq \mathcal{I}$ is called a *k-itemset*. A transaction t_i contains an itemset X in I, if $X \subseteq t_i$. The support of an itemset X, noted $\sigma(X)$, is the percentage of transactions contained in $\mathcal{T}$ in which X is a subset :

$$support(X) = \frac{|\{t \in T | X \subseteq t\}|}{|t \in T|}$$

An itemset is frequent if the support $\sigma(X) \geq minsup$, where *minsup* is the user-specified minimum support. An association rule is an implication such as $X_1 \mapsto X_2$, where X_1 and X_2 are itemsets with $X_1, X_2 \subseteq I$ and $X_1 \cap X_2 = \varnothing$. The support of an association rule r: $X_1 \mapsto X_2$ is equal to the support of the union of itemsets which establish it (Agrawal et al., 1993) :

$$support(r) = support(X_1 \cup X_2)$$

The confidence of an association rule is the conditional probability that the transaction contains X_2 knowing X_1 :

$$confidence(r) = \frac{support(r)}{support(X_1)}$$

Rules that satisfy both a minimum support threshold and a minimum confidence threshold are called strong rules. The following subsection presents automatic and user-driven approaches.

2.2 Automatic approach vs user-driven approach

Association rules mining is an automatic task where the user appears at the beginning and at the end within the process. First of all, he determines the support and the confidence of the algorithm. Once the mining ended, he rates the results obtained. The main problem of this method is the large number of generated rules (Kuntz et al., 2000). To solve it, the user-driven approach applied to association rules mining was proposed. The last two stages of KDD: datamining and post-processing was grouped. The user is at the heart of mining and he can drive it throughout the process (Kuntz et al., 2000). This is the main difference with automatic approach. The main problem of this approach is to present comprehensive and fast results so that the user does not waste his time analyzing current results. Various works on rules filtering by quality criteria were proposed (Ohsaki et al., 2004) but the experience of the user is not exploited during all the KDD process. We present related works in the next subsection.

2.3 Related works

All techniques currently developped have a common purpose which is to discover association rules in databases. One of the major problems of association rules mining is the large number of patterns which are generated. It is difficult for the expert to identify those association rules that are interesting for him. To help the expert to choose among these patterns, several works were proposed on association rules filtering according to their interestingness (Liu et al., 1996). Other works are proposed on hierarchical association rules mining. Let $\mathcal{I}$ and $\mathcal{T}$ as previously presented, and let $\mathcal{G}$, a taxonomy or hierarchical tree. A taxonomy is a directed acyclic graph (DAG) on the items in I where items are the leaves and where edges are inheritance relation. This approach allows to generate multi-level association rules (Han and Fu, 1995; Hipp et al., 1998; Srikant and Agrawal, 1997; Srikant et al., 1997; Tseng, 2001).

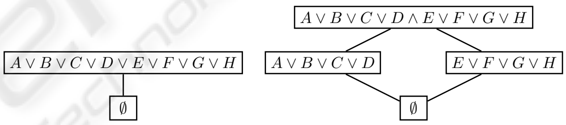
In these works, there are two different kind of relevance measures. The first are called objective measures (Silberschatz and Tuzhilin, 1995). These measures are data-oriented (Bayardo and Agrawal, 1999; Tan et al., 2002; Hilderman and Hamilton, 2001). Several of these works are efficient to discover the best rules or to estimate the best rules (Morimoto et al., 1998) thanks to one of these objective measures but they are limited. Indeed, there are no measures which are able to treat efficiently random problems. They are successful in particular contexts. Furthermore, they do not allow yet to rate patterns quality (Liu et al., 1996). A comparison of these measures was done in (Ohsaki et al., 2004).

The second measures are called subjective measures. These second measures are user-oriented (Padmanabhan and Tuzhilin, 1998; Liu et al., 1996). The expert's tacit knowledge, in a research domain, are taken into account. Contrary to objective measures which are numerous, subjective measures is a recent research domain. We can count a dozen measures and split them into three sub-groups: unexpected (Matheus et al., 1996; Klemettinen et al., 1994; Silberschatz and Tuzhilin, 1996; Liu et al., 1996; Liu et al., 1997; Padmanabhan and Tuzhilin, 1999; Liu et al., 2000; Shekar and Natarajan, 2002), actionability (Klemettinen et al., 1994; Piatetsky-Shapiro and Matheus, 1994; Freitas, 1999) and anticipation (Roddick and Rice, 2001). These different measures exploit the knowledge of the expert to confront them with the results of datamining. The aim is to highlight useful and more or less unexpected knowledge. In this paper, we focus on this second family of measures. The conception of subjective measure is a difficult task for several reasons. First, experts have not common interests for a common research domain. Secondly, given a database, and a knowledge set, different experts may be interested in different subsets of this explicit knowledge. Finally, the expert's conclusions can vary depending on time and situation. These various points show how complex the problem is. We present in the next section, our new method based on a combination of both user-driven and hierarchical association rules mining approaches.

3 HYBRID ALGORITHM FOR ASSOCIATION RULES MINING

3.1 Algorithm details

First of all, item taxonomy must be created in cooperation with the expert of the domain. The aim is to develop the taxonomy by grouping items possessing common roots. The rule will be specialized at the following level according to the expert's choices. The current association rule mining consists in building a lattice of frequent itemsets. In our example, the lattice is created depending on level 4 of the taxonomy, that is the lowest level. We are working on taxonomy level. The starting level is the highest one (cf Fig 1 (a)).



(a) Level 1 (b) Level 2

Figure 1: Lattice generated at levels 1 and 2

But, association rules are not generated with a single itemset. Consequently, the starting level will always be level 2 (cf Fig 1 (b)). Thenceforth, there are two possibilities. First of all, association rules are not generated. In this case, mining is restarted from the next level. Secondly, rules are generated depending on level 2 and proposed to the expert. He selects relevant rules for him. The taxonomy is pruned according to this selection. The corresponding itemsets are kept

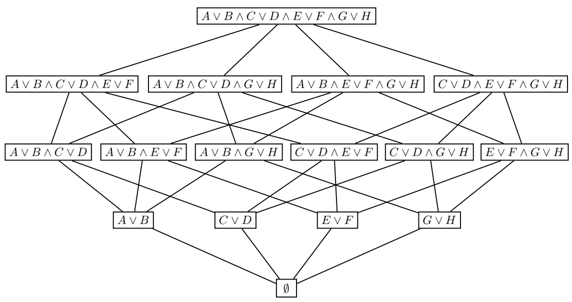


Figure 2: Lattice generated at level 3

in the taxonomy, and other itemsets and their inherited items are pruned (Hipp et al., 1998). We consider, for our example, that for this level, the taxonomy was not pruned. Mining is then achieved at the next level (cf Fig 2). As previously, rules are generated at level 3 (cf Fig 2). For instance, the expert selects one rule "If $A \vee B \wedge G \vee H \mapsto E \vee F$". The itemset $C \vee D$ does not appear in this rule, and will then be pruned from the taxonomy as well as the inherited items C and D (cf Fig 3) (Hipp et al., 1998).

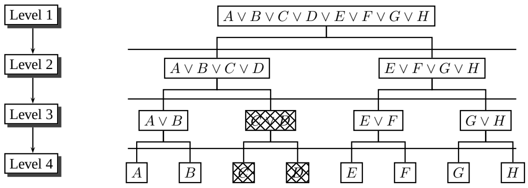


Figure 3: Item taxonomy

If no relevant aggregated rules are found, mining is restarted from the last level of the taxonomy, as classical association rules mining. But, for our example, there is no more than 2^8 at the beginning but 2^6 thanks to the taxonomy pruning. The main advantage of this method is the variation of space and time complexities at the search level. If we increase this search level, running time are going to decrease. The useless or coarse association rules will be discovered from the beginning and they are not developed in the next levels. The number of generalized rules will be more or less low. This algorithm uses a fixed support. This approach allows to generate mono-level rules. We present our algorithm in the next subsection.

3.2 Search Hierarchical Association Rules for Knowledge (SHARK)

SHARK algorithm is described hereafter (see algorithm 1). The Update function is not detailed because it is a simple prunning of the items for the current level. The RuleGeneration() function is not detailed too in this paper because this function doesn't present a new idea. It runs Apriori which is a parameter that can be replaced by any other rule generation algorithm. Starting at level i, the algorithm updates the set of valid items at this level. Given the $minsup$, a search for frequent itemsets is achieved with the set of selected items. A procedure RuleGeneration is run, given a confidence thereshold, to generate a rules set. This rules set is analyzed by the expert which selects relevant ones. Items which are not present among relevant rules are removed from the items list. The procedure restarted at the next level $i + 1$ until the last level is reached. We have implemented a datamining platform, Lminer, in which both Apriori (Agrawal et al., 1996) and SHARK are included. SHARK is an extension of Apriori algorithm that integrates our hybrid approach. We present in the next section experimental results.

Algorithm 1: SHARK Algorithm

function *AssociationRuleGenerate(Set of items $\mathcal{I}$, Database $\mathcal{T}$, Taxonomy $\mathcal{G}$)*
Data: Set containing all the items,
Database containing all the transactions,
Item taxonomy
Result: Set of association rules F_{ra}
begin
  Level=1;
  while *NextLevel==OK and user!=END* **do**
    // Increase level
    Level++;
    // Updating the set of items depending on the //search level
    $\mathcal{I} \leftarrow$ Update($\mathcal{I}$,Level);
    // Frequent itemsets search
    $F_g \leftarrow$ FrequentSearch($\mathcal{I}$,$\mathcal{T}$,$\mathcal{G}$,minsup);
    // Generation of association rules
    $F_{ra} \leftarrow$ RuleGeneration(F_g,$\mathcal{G}$,minconf);
    // Presentation of rules to the expert
    $F_{raFinal} \leftarrow$ ChooseRule(F_{ra});
    // Pruning taxonomy
    PruneTaxonomy($\mathcal{G}$,$F_{raFinal}$);
    // End test
    if *Level==MaxLevel* **then**
      NextLevel==NotOK;
  return $F_{raFinal}$;
end

4 RESULTS

Our purpose is to include a hierarchical mining and a user-driven approach. We don't want propose a new vizualisation methodology. We developed an easy-to-use graphical interface for our experiments. Indeed, extracting nuggets is very difficult when the relevant information is hidden in a large amount of data. Various works already exist to help expert analysis (Klemettinen et al., 1996; Liu et al., 1999). These two works were completed by several works for rules exploration (Blanchard et al., 2003; Ben-Yahia and Mephu-Nguifo, 2004). A set of association rules is proposed to the expert at each level. He chooses the rules which he finds relevant just with a mouse click then he runs the mining to the following level with a button and so on.

First of all, we generated the taxonomy with the experts. The example shows a balanced tree (cf Fig 3) but it is not always balanced. It has no influence on the algorithm running. When the leaves are not on the last level of the taxonomy, the algorithm selects all the leaves on the previous levels. Our taxonomy is composed of 114 nodes and 57 leaves. To estimate the rules, a graphical interface was developed to hold some specific items. This tool allows to vary several metrics: absolute or relative support rate, confidence rate, number of rules, number of items of the antecedent and the conclusion and to select specific items of the antecedent and the conclusion. Thanks to this tool, the number of rules was reduced from 2445 to few dozens according to the selection of three experts working together. It is the reason which explains the short expertise time. We tested our data with Apriori (automatic approach) and with Shark (user-driven approach). Our benchmark is composed of 57 items (the last level of the taxonomy) and 400 000 transactions. We work with a pentium III 700 Mhz with 128 Mo RAM. The metrics $minsup$ and $minconf$ are fixed by the experts. Our results are presented in table 1.

Table 1: Experiment results

	Nb rules	Calcul	Sort	Expert	Select
Apriori	2445	1140s	56	**600s**	**10**
Shark (L1)	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
Shark (L2)	5	1s	2	30s	2
Shark (L3)	546	70s	16	60s	6
Shark (L4)	2113	1080s	36	300s	8
Shark	2113	1151s	36	**390s**	**8**

Apriori generates 2445 rules in 1140 seconds. This time does not include the expertise time. It is necessary to add about 600 seconds for a total of 1740 seconds. Shark algorithm works on more and less aggregated data. At the first level, no rule is generated. The mining begins on the second level. Shark generates 5 rules in 1 second and 30 seconds of expertise time. At the first level, 546 rules generated in 70 seconds and 60 seconds of expertise time. Finally, at the last level, as Apriori algorithm, 2113 rules are generated in 1080 seconds and 300 seconds of expertise time. It is necessary to note that this results vary according to the material used.

The total running time of the algorithm is 1151 seconds and 1140 seconds for Apriori. Shark is longer than Apriori. It is because the experts uses a interactive graphical interface. A mining on a level depends on an interaction between the experts and the interface. As long as he does not click on a button to calculate the next level, mining is stopped. It is the reason why the running time of fewer rules are longer with Shark algorithm. Anyway, our main problem is not the running time but the expertise time. The total expertise time with Shark (390s) is shorter than Apriori (600s). The number of rules with SHARK (8 rules) is smaller than the number obtained with Apriori (10 rules). The two rules was pruned according to experts' aims. The confidence of these two rules were very low, and the experts qualified them as noise among the initial set of selected rules. Furthermore in our experiments, the experts was able to obtain a gain of 35% of the expertise time when using SHARK, while the number of selected rules remains almost significantly identical. However, this observation should be validated on different other applications and with different experts. In fact, the loss of rules during the whole process had to be minimized to increase the efficiency of the method. And this is also linked to expert's choices at each level.

5 CONCLUSIONS AND FUTURE WORKS

The user-driven approach allows to moderate two problems. On the one hand, the limitations of expert reflexion and on the other hand, the problem of data volume which does not allow to obtain quickly relevant results by automatic approach (Silberschatz and Tuzhilin, 1996). In this paper, we have presented our work about a hybrid method based on a hierarchical association rules mining and a user-driven approach. The main idea is to decrease the expertise time and the number of generated rules. Hierarchical association rule mining is used to decrease expertise time. Mining is performed level by level and is oriented by expert's choices. The direct consequence is that the number of generated rules is small and these rules are better targeted because the expert directs the mining during all the process.

The experiments show that hybridizing the hierarchical association rules mining and the user-driven approach is interesting especially in our application case of banking marketing. Indeed, the Shark methodology allowed to decrease the total expertise time and the number of rules. A KDD process must be steered according to the initial purposes of the expert. Thus, his role is really essential before, after, but especially during the process. One limitation of our approach is that the number of generated rules remains nevertheless large. We reduce the problem but it needs to be improved. Another limitation is that we define a taxonomy with simple inheritances. We don't treat the multiple inheritances as in (Srikant and Agrawal, 1997).

ACKNOWLEDGMENTS

This work has been partly supported by the "Centre National de la Recherche Scientifique" (CNRS), the "Association Nationale de la Recherche Scientifique" (ANRT), the "IUT de Lens" and the "Université d'Artois". A special thanks to the marketing service of the "Caisse d'Epargne du Pas de Calais".

REFERENCES

Agrawal, R., Imielinski, T., and Swami, A. N. (1993). Mining association rules between sets of items in large databases. In Buneman, P. and Jajodia, S., editors, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, D.C.

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., and Verkamo, A. I. (1996). Fast discovery of association rules. In *Advances in knowledge discovery and data mining*, pages 307–328. American Association for Artificial Intelligence.

Bayardo, R. J. and Agrawal, R. (1999). Mining the most interesting rules. In *Proc. of KDD'99*, pages 145–154, San Diego, USA.

Ben-Yahia, S. and Mephu-Nguifo, E. (2004). Emulating a cooperative behavior in a generic association rule visualization tool. In *Proc. of ICTAI'04*, Boca Raton, Florida, USA.

Blanchard, J., Guillet, F., and Briand, H. (2003). Exploratory visualization for association rule rummaging. In *Proceedings of the 4th International Workshop on Multimedia Data Mining MDM/KDD'03*, Washington.

Fayyad, U., Piatetsky-Shapiro, G., and Smyth, P. (1996). From data mining to knowledge discovery: An overview. *In Advances in Knowlegde Discovery and Data Mining, pages 1-34*.

Freitas, A. A. (1999). On rule interestingness measures. *Knowledge-Based Systems journal*, 12(5-6):309–315.

Han, J. and Fu, Y. (1995). Discovery of multiple-level association rules from large databases. In *Proc. of VLDB'95*, pages 420–431, Zürich, Switzerland.

Hilderman, R. J. and Hamilton, H. J. (2001). *Knowledge discovery and measures of interest*. Kluwer Acaedemic Publishers.

Hipp, J., Myka, A., Wirth, R., and Güntzer, U. (1998). A new algorithm for faster mining of generalized association rules. In *Proc. of PKDD '98*, pages 74–82, Nantes, France.

Klemettinen, M., Mannila, H., Ronkainen, P., Toivonen, H., and Verkamo, A. I. (1994). Finding interesting rules from large sets of discovered association rules. In *Proc. of CIKM'94*, pages 401–407. ACM Press.

Klemettinen, M., Mannila, H., and Toivonen, H. (1996). Interactive exploration of discovered knowledge : A methodology for interaction and usability studies. Technical report, University of Helsinki.

Kuntz, P., Guillet, F., Lehn, R., and Briand, H. (2000). A user-driven process for mining association rules. In *Proc. of PKDD'00*, pages 160–168, Lyon, France.

Liu, B., Hsu, W., and Chen, S. (1997). Using general impressions to analyze discovered classification rules. In *Proc. of KDD'97*, pages 31–36. AAAI Press.

Liu, B., Hsu, W., Chen, S., and Ma, Y. (2000). Analyzing the subjective interestingness of association rules. *IEEE Intelligent systems*, 15(5):47–55.

Liu, B., Hsu, W., Mun, L.-F., and Lee, H.-Y. (1996). Finding interesting patterns using user expectations. volume 11, pages 817–832.

Liu, B., Hsu, W., Wang, K., and Chen, S. (1999). Visually aided exploration of interesting association rules. In *Proc. of PAKDD'99*, pages 380–389, Beijing, China.

Matheus, C. J., Piatetsky-Shapiro, G., and McNeill, D. (1996). Selecting and reporting what is interesting : the kefir application to heathcare data. *In Advances in Knowlegde Discovery and Data Mining*, pages 495–516.

Morimoto, Y., Fukuda, T., Matsuzawa, H., Tokuyama, T., and Yoda, K. (1998). Algorithms for mining association rules for binary segmentations of huge categorical databases. In *Proc. of VLDB'98*, pages 380–391, New York City, New York, USA.

Ohsaki, M., Kitaguchi, S., Okamoto, K., Yokoi, H., and Yamaguchi, T. (2004). Evaluation of rule interestingness measures with a clinical dataset on hepatitis. In *Proc. of PKDD'04*, pages 362–373, Pisa.

Padmanabhan, B. and Tuzhilin, A. (1998). A belief-driven method for discovering unexpected patterns. In Press, A., editor, *Proc. of KDD'98*, pages 94–100, New York, USA.

Padmanabhan, B. and Tuzhilin, A. (1999). *Unexpectedness as a measure of interestingness in knowledge discovery*. Decision support system, 27.

Piatetsky-Shapiro, G. and Matheus, C. J. (1994). The interestingness of deviations. In *Proc. of KDD'94*, pages 25–36, Seattle, USA.

Roddick, J. F. and Rice, S. (2001). What's interesting about cricket? on thresholds and anticipation in discovrerd rules. *SIKKDD Explorations*, 3(1):1–5.

Shekar, B. and Natarajan, R. (2002). A fuzzy-graph-based approach to the determination of interestingness of association rules. In *Proc. of PAKM'02*, pages 377–388.

Silberschatz, A. and Tuzhilin, A. (1995). On subjective measures of interestingness in knowledge discovery. In *Proc. of KDDM'95*, pages 275–281, Menlo Park, Californie, USA.

Silberschatz, A. and Tuzhilin, A. (1996). User-assisted knowledge discovery : how much should the user be involved. In *Proc. of DMKD'96*, Montreal, Canada.

Srikant, R. and Agrawal, R. (1997). Mining generalized association rules. *Future Generation Computer Systems*, 13(2–3):161–180.

Srikant, R., Vu, Q., and Agrawal, R. (1997). Mining association rules with item constraints. In *Proc. of KDD'97*, pages 67–73.

Tan, P.-N., Kumar, V., and Srivastava, J. (2002). Selecting the right interestingness measure for association patterns. In *Proc. of KDD'02*, pages 32–41. ACM Press.

Tseng, S. (2001). Mining association rules with interestingness constraints in large databases. *In International Journal of Fuzzy Systems*, 3(2).