Generalized Knowledge Discovery from Relational Databases

Yu-Ying Wu[†], Yen-Liang Chen[†], and Ray-I Chang[††]

[†]Department of Information Management, National Central University, 300 Jhongda Road, Jhongli, Taiwan

[††]Department of Engineering Science and Ocean Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan

Summary

The attribute-oriented induction (AOI) method is a useful tool for data capable of extracting generalized knowledge from relational data and the user's background knowledge. However, a potential weakness of AOI is that it only provides a snapshot of generalized knowledge, not a global picture. In addition, the method only mined knowledge from positive facts in databases. Rare but important negative generalized knowledge can be missed. Hence, the aim of this study is to proposal two novel mining approaches to generate multiple-level positive and negative generalized knowledge. The approaches discussed in this paper are more flexible and powerful than currently utilized methods and can be expected to have wide applications in diverse areas including e-commerce, e-learning, library science, and so on.

Key words:

Data mining, Attribute-oriented Induction, Knowledge discovery, Multiple-level mining, Negative pattern

1. Introduction

Data mining is useful in various domains, such as market analysis, decision support, fraud detection, business management, and so on [1, 2]. Many data mining approaches have already been proposed in discovering useful knowledge. One of the most important of these approaches is the attribute-oriented induction (AOI) method which was first introduced by Cai et al. in 1990 [3].

AOI is a data mining technique to extract generalized knowledge from relation data and user's background knowledge. The essential background knowledge applied in AOI is concept hierarchy associated with each attribute in the relational database [4]. A concept hierarchy often refers to as domain knowledge, and stores relationships of specific concepts and generalized concepts. The generalization process is performed by either attribute removal or concept hierarchy ascension, and controlled by two parameters: the attribute generalization threshold (*Ta*) and the relation generalization threshold (*Tr*) [1, 5]. The attribute threshold specifies the maximum number of distinct values of any attribute that may exist after generalization, and the relation threshold gives an upper bound on the number of the generalized tuples that remain after the generalization process [6]. Given the specific thresholds, these two parameters can be applied to generate a set of generalized tuples to describe the target relation. Integrating with the concept hierarchies, the AOI methods can induce multi-level generalized knowledge, which can provide good decision making support.

There is no doubt that AOI technique is very useful for inducing the general characteristics of an input relation, and has been applied in many areas such as spatial patterns [7, 8], medical science [9, 10], intrusion detection [11] and strategy making [12]. However, the AOI method has a serious weakness. That is, the AOI method only provides a snapshot of the generalized knowledge, not a global picture. If we set different thresholds, we will obtain different sets of generalized tuples that also describe the major characteristics of the input relation. If a user wants to know the global picture of induction, he or she must try different thresholds repeatedly. That is a time-consuming and tedious work. It is therefore important to provide an efficient algorithm to generate all interesting generalized tuples at a time.

In addition, as we can see from the applications, existing AOI approaches only mined knowledge from positive facts in databases. In real-world, negative generalized knowledge which is unknown, unexpected, or contradictory to what the user believes is more novel, meaningful and interesting than positive facts to the user. Discovering negative tuples from relational database can reveal more interesting knowledge. For example, a bank database, may tell us that customers in high level positions seldom return the loan exceed the time limit. Even in the specific concept, the manager positions which are the subset of high level positions did not return the loan exceed the time limit. This example illustrates that negative generalized knowledge in different level can provide valuable information and help different level managers in devising better decision making.

From the above we proposed two novel generalized knowledge induction methods to generate multiple-level positive and negative generalized knowledge. The experimental results show that the two methods are efficient and scalable for use in relational databases. Moreover, the proposed pruning strategies incorporated into the algorithm can make the computational efficiency satisfactory.

The remaining of the paper is organized as follows. In Section 2, we review the AOI method and related work. In

Manuscript received June 5, 2009.
Manuscript revised June 20, 2009.

Section 3, we propose two algorithms to generate all interesting generalized tuples from relational databases. Section 4 shows the experiment results. Finally, the conclusions are drawn in Section 5.

2. Overview of Attribute-Oriented Induction

AOI is a set-oriented database mining method which transforms data stored in database relations into more general information on an attribute by attribute basis [13, 14]. The input of the method includes a relation table and a set of concept trees (concept hierarchies) associated with the attributes of the table. The concept hierarchies represent necessary background knowledge which controls the generalization process. Different levels of concepts are often organized into taxonomy of concepts. The concepts range from a single, most generalized concept at the root to the most specific concepts corresponding to the specific values of attributes in the database [5, 6].

The idea of AOI is to perform generalization based on the examination of the number of distinct values of each attribute in the relevant set of data. As mentioned in Section 1, there are two common parameters to control the generalization process. The attribute generalization threshold (*Ta*) specifies the maximum number of distinct values of an attribute. If the number of distinct values in an attribute is greater than the attribute threshold, further attribute removal or attribute generalization should be performed.

Through attribute removal or attribute generalization, the AOI algorithm then apply the second parameters, the generalized relation threshold (*Tr*), to further aggregate the generalized relation. If the number of distinct tuples in the generalized relation is greater than the threshold, further aggregation should be performed. Aggregation is performed by merging identical, generalized tuples, and accumulating their respective counts [1]. The following example illustrates a resulting generalized relations using AOI method.

Table 1. The result after generalizing the car data with AOI, *Ta*=6, *Tr*=6.

rid	Manufacturer	Model	Engine Displacement	Price	Vote
1	USA	Any	Middle	Valued	4
2	USA	Any	High	Valued	4
3	Korea	Any	Low	Cheap	3
4	Japan	Any	High	Economic	3
5	Korea	Any	High	Economic	2
6	Germany	Any	High	Expensive	2

Based on the AOI approach, researchers have proposed various extensions. Carter and Hamilton proposed more efficient methods of AOI [14]. Cheung proposed a rule-based conditional concept hierarchy, which extends traditional approach to a conditional AOI and thereby allows different tuples to be generalized through different paths depending on other attributes of a tuple [15]. Hsu extended the basic AOI algorithm for generalization of numeric values [16]. Chen and Shen proposed a dynamic programming algorithm, based on AOI techniques, to find generalized knowledge from an ordered list of data [6]. Several independent groups of researchers have investigated applications of a fuzzy concept hierarchy to AOI [17-20]. A fuzzy hierarchy of concepts reflects the degree which a concept belongs to its direct abstract. In addition, more than one direct abstract of a single concept is allowed during fuzzy induction. Other researchers integrated AOI with other applications to generalize complex database [7, 8, 21]. To the best of our knowledge, all previous research does not address the problems studied in the paper.

Moreover, the above approaches, all concentrate on mining positive generalized knowledge. The study of mining negative generalized tuples has not been addressed. The similar researches are only the negative association rules mining methods [22-25]. Savasere [23] and Yuan [24] combines previously discovered positive associations with domain knowledge and only part of the rules are focused on to reduce the scale of the problem. The algorithms generate negative rules based on a complex measure of rule parts and are mainly restricted to relative negative association rules compared with other sibling itemsets. Wu [25] and Antonie [22] presented an Apriori-based framework for mining both positive and negative association rules. The algorithms proposed can hardly guarantee to generate a complete set of valid rules.

In this paper, our approaches take a fundamentally different from those discussed so far. The goal of this paper is to develop two efficient mining methods for discovering positive and negative knowledge across different levels of taxonomies from relational databases.

3. Induction Methods

In this section, two novel generalized knowledge induction approaches, generalized positive knowledge induction (GPKI) and generalized negative knowledge induction (GNKI), are proposed to remedy the drawbacks of the traditional AOI method. The first method, GPKI, employs the support concept and multiple mining technique to generate the global positive knowledge at one time. Different from the positive knowledge generalization, The second method, GNKI, discover the rare but important tuples. It empoys two closure property for saving memory space and speed up the computation. The we discuss in detail the algorithm designed for inducing all generalized tuples.

3.1 Preprocessing

The general idea of AOI is to first collect the task-relevant data using a relational database query. Our approaches are the same as the AOI algorithm. Fig. 1 shows the preprocessing process of the proposed methods.

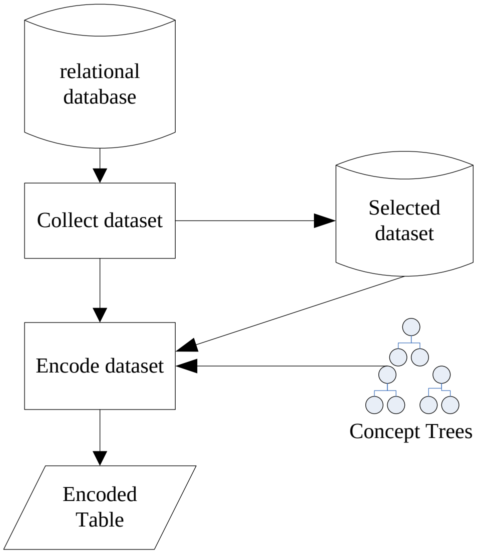

Fig. 1 The preprocessing process.

The input of the algorithm is an entire relational database containing enormous tuples. The first process is to collect the data set that is relevant to the learning task using relational database operations, e.g., projection or selection. The selected task-relevant datasets are stored in a database. Next, the encoding process uses the selected datasets and the concept hierarchies to transform the original tuples to a hierarchy-information encoded table, which stores the encoding code of the corresponding value of attribute. The encoding code can be used as the input of generalized algorithms.

3.2 Mining Positive Tuples

As mentioned in section 1, traditional AOI method can only mining a snapshot of the generalized knowledge, not a global picture. Hence, the goal of the generalized positive knowledge induction (GPKI) algorithm is to proposal a novel method to discover all multi-level generalized tuples at a time. Fig. 2 illustrates the generalization procedure. The input of the GPKI is the encoded table generating by the preprocessing process. The encoded table can facilitate the frequent discovering process efficiently.

The idea for discovering all frequent generalized tuples is extended from the method for mining multiple-level association rules [26] and employs the concept of multiple minimum supports proposed by Liu [27]. The process first scan the encoded table to find all potential frequent 1-valuesets and all frequent 1-valuesets at all levels. Let the C_1 and P_1 denote the set of candidate 1-valuesets and the set of potential frequent 1-valuesets respectively. The L_k is the set of frequent k-valuesets. Next, the algorithm generates the k-valuesets at all levels using P_1 and L_k. The k-valuesets generation is repeated until k is equal to the number of attributes in tuple. Finally, to gather the all k-valuesets, we can obtain the overall encoding general tuples.

Because of the general tuples are not all interesting enough to be presented to users. We must suggest an interest measure to filter uninteresting tuples. The final step is to transform all interesting encoding generalized tuples to original tuples. Finally, the output consists of interesting generalized tuples learned from a relational database.

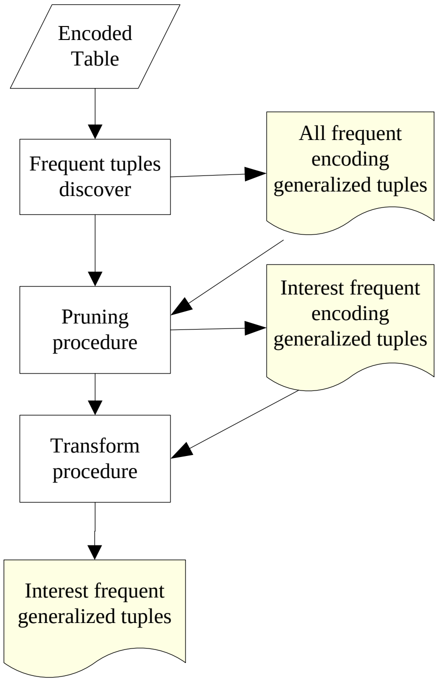
Encoded Table
Frequent tuples discover
All frequent encoding generalized tuples
Pruning procedure
Interest frequent encoding generalized tuples
Transform procedure
Interest frequent generalized tuples

Fig. 2 The processes of the GPKI.

3.2 Mining Negative Tuples

The generalized negative knowledge induction (GNKI) algorithm mines multiple-level negative generalized tuples from relational database. The input of the GNKI is the same as the GPKI. However, different from the GPKI, the GNKI is to discover the rare but important tuples called negative tuples. In the previous AOI researches, discovering negative generalized knowledge is ignored. In this study, we proposed a novel algorithm to mine multi-level negative generalized knowledge. The algorithm can be summarized into the following steps, as shown in Fig. 3.

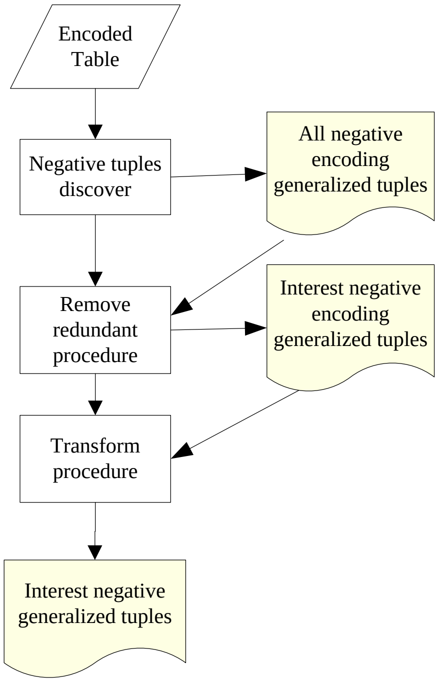
Encoded Table
Negative tuples discover
All negative encoding generalized tuples
Remove redundant procedure
Interest negative encoding generalized tuples
Transform procedure
Interest negative generalized tuples

Fig. 3 The processes of the GNKI.

The generalization of negative valuesets is a difficult task and the computational load will be heavy. As we have seen, there can be an exponential number of infrequent valuesets in a database, and only some of them are useful for mining negative generalized knowledge. Therefore, filtering and pruning strategy is critical to efficient generation. In this study, we analyze the properties of all valuesets, and suggest two pruning strategy to efficiently generate global negative generalized knowledge. The first step in the process is to scan the encoded table, and find the set of all values (*CV*) and the set of frequent 1-valuesets (PL_1) for all levels. Let PL_k be the set of potential *k*-valuesets. All *k*-valuesets are generated using PL_{k-1} and *FV*. The *k*-valueset generation is repeated until *k* is equal to the number of attributes in the tuple. Finally, we obtain all negative generalized tuples by uniting all *k*-valuesets.

The negative generalized tuples discovered by the first process include multiple level valuesets. The tuples may have many redundancies. In addition, not every generalized tuple is worth enough to be presented to users. Hence, we must suggest an effective redundant procedure to remove all unnecessary negative tuples. The final step is to transform all nonredundant negative generalized tuples. Then, we output the final tuples.

4. Examples and Experiments

In this section, we perform a simulation study to empirically evaluate the performance of the proposed method. The algorithms were implemented in Java language and tested on a PC with an Intel Pentium D 2.8GHz processor and 2GB main memory under the Windows XP operating system. A synthetic data set of car information data is used to carry out the experiments which are made in the same running environment. To make the time measurements more reliable, no other application was running on the machine while the experiments were running.

The first experiment evaluates the run time of the GPKI algorithm with varied data size (the numbers of tuples) from 5,000 to 50,000, and used a constant set of relevant thresholds. Besides, this evaluation is run by setting three different numbers of attributes, i.e., 4 attributes (attr-4), 5 attributes (attr-5) and 6 attributes (attr-6). The results are illustrated in Fig. 4. From this figure, we can see that time required by three cases increases as the number of tuples increases. Moreover, the more attributes are given in experiments, the more run time be required. This is because the GPKI algorithm generates the candidate valuesets by combining distinct attribute values recursively. Hence, the number of attributes has significant effect for run time. And the gap of the run time among attributes is larger and larger following the data size increases.

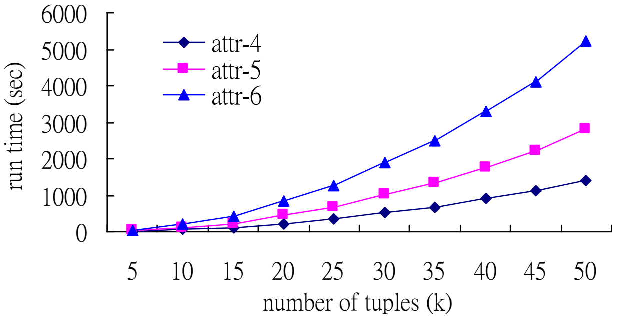

Fig. 4. Run time of GPKI algorithm with different data size.

Because of the GPKI algorithm discovers multi-level generalized knowledge. In different level, we must set the different filter threshold. Next, we study how the minimal level support thresholds influence the run time of the GPKI algorithm. To this end, we give five sets of the thresholds and fix the number of tuples as 25000, the attributes as 6. The thresholds are set as follows: (1) 9%, 6%, 3%, (2) 8%, 6%, 4%, (3) 15%, 10%, 5%, (4) 10%, 8%, 6% and (5) 15%, 12%, 10%. Fig. 5 shows the performance curve. Time required by the algorithm increases as the threshold values decreases. This result is because of the thresholds influence the number of candidate values which were used to combine as longer valuesets. The more larger threshold values, the less candidate values.

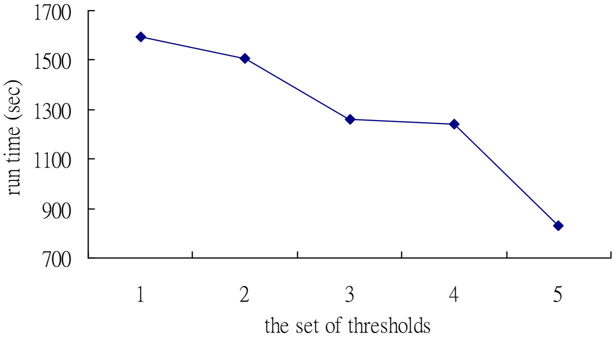

Fig. 5 Run time of GPKI algorithm with different thresholds.

Third, we show how the numbers of generalized tuples changes as the level threshold values and the data sizes be changed. To do so, we let the level threshold values and data sizes are fixed. Note, because of not every generalized tuple is interesting enough to be presented to users. In this experiment, we must set the interest measure as 1.1. Fig. 6 shows the result with different set of thresholds. From this figure, one can see that the number of generalized tuples decreases as the threshold values get larger. This result is quite reasonable, because the thresholds are used to filter out the valuesets, the smaller thresholds can keep the more valuesets. On the other hand, setting the larger thresholds can filter out more valuesets.

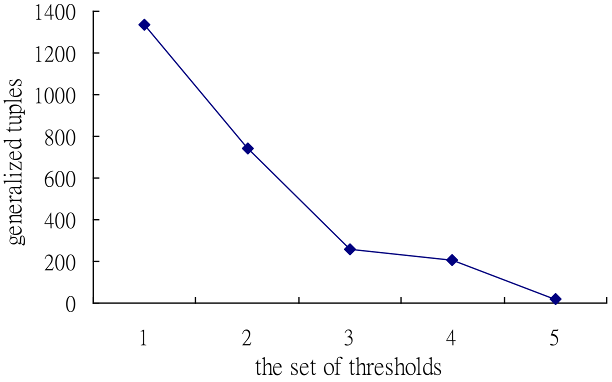

Fig. 6 The number of generalized tuples changed when the thresholds are changed.

We also concern the number of generalized tuples changed when the data sizes be changed. The results are shown in Fig. 7. In the experiment, we fix the parameters as follows: the thresholds as 15%, 10%, 5%, attributes as 5 and interest measure as 1.1. From this result, we found that the number of generalized tuples is similar. Since the different data sizes are extracted from same dataset, the tuples may have the similar property. When the thresholds are set the same, the number of generalized tuples that generated by the algorithm are similar.

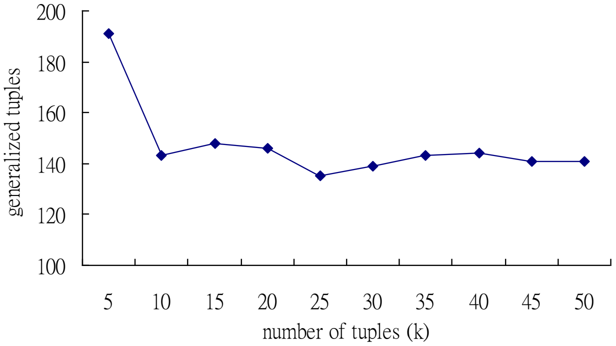

Fig. 7 The number of generalized tuples changed when the data sizes are changed.

5. Conclusion

We present two efficient methods, generalized positive knowledge induction and generalized negative knowledge induction, for generating all interesting generalized knowledge at one time. The algorithms provide a simple and efficient way for knowledge generalization from a relational database. A dataset is used for experiment. The results show that the proposed method is efficient and scalable for relational databases. It is interesting to extend the flexibility of methods by using domain generalization graphs rather than concept trees or using fuzzy concept trees rather than crisp concept trees.

References

[1] J. Han and M. Kamber, *Data mining: Concepts and techniques*, *second edition* (Morgan Kaufmann, New York, 2006).

[2] S.Y. Chen and X. Liu, The contribution of data mining to information science, *Journal of Information Science* 30(6) (2004) 550-8.

[3] Y. Cai, N. Cercone and J. Han, An attribute-oriented approach for learning classification rules from relational databases. *Proceedings of the Sixth International Conference on Data Engineering* (Washington, DC, 1990) 281-8.

[4] M.S. Chen, J. Han and P.S. Yu, Data mining: An overview from a database perspective, *IEEE Transactions on Knowledge and Data Engineering* 8(6) (1996) 866-83.

[5] J. Han, Y. Cai and N. Cercone, Knowledge discovery in databases: An attribute-oriented approach. *Proceedings*

of the 18th International Conference on Very Large Data Bases (Vancouver, Canada, 1992) 547-59.

[6] Y.-L. Chen and C.-C. Shen, Mining generalized knowledge from ordered data through attribute-oriented induction techniques, *European Journal of Operational Research* 166(1) (2005) 221-45.

[7] E.M. Knorr and R.T. Ng, Extraction of spatial proximity patterns by concept generalization. *Second International Conference on Knowledge Discovery and Data Mining* (Portland, Oregon, 1996) 347–50.

[8] L.Z. Wang, L.H. Zhou and T. Chen, A new method of attribute-oriented spatial generalization. *Proceedings of 2004 International Conference on Machine Learning and Cybernetics* (Shanghai, China, 2004) 1393-8.

[9] Q.-H. Liu, C.-J. Tang, C. Li, Q.-W. Liu, T. Zeng and Y.-G. Jiang, Traditional chinese medicine prescription mining based on attribute-oriented relevancy induction, *Journal of Computer Applications* 27(2) (2007) 449-52.

[10] S. Tsumoto, Knowledge discovery in clinical databases and evaluation of discovered knowledge in outpatient clinic, *Information Sciences* 124(1-4) (2000) 125-37.

[11] J. Kim, G. Lee, J. Seo, E. Park, C. Park and D. Kim, An alert reasoning method for intrusion detection system using attribute oriented induction. *Information Networking: Convergence in Broadband and Mobile Networking, ICOIN 2005* (Jeju Island, Korea, 2005).

[12] S.T. Li, L.Y. Shue and S.F. Lee, Business intelligence approach to supporting strategy-making of isp service management, *Expert Systems with Applications* 35(3) (2008) 739-54.

[13] J. Han and Y. Fu, Exploration of the power of attribute-oriented induction in data mining, in Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P. and Uthurusamy, R. (eds.), *Knowledge discovery and data mining* (AAAI/MIT Press, Cambridge, Mass., 1996) 399-421.

[14] C.L. Carter and H.J. Hamilton, Efficient attribute-oriented generalization for knowledge discovery from large databases, *IEEE Transactions on Knowledge and Data Engineering* 10(2) (1998) 193-208.

[15] D.W. Cheung, H.Y. Hwang, A.W. Fu and J. Han, Efficient rule-based attribute-oriented induction for data mining, *Journal of Intelligent Information Systems* 15(2) (2000) 175-200.

[16] C.-C. Hsu, Extending attribute-oriented induction algorithm for major values and numeric values, *Expert Systems with Applications* 27(2) (2004) 187-202.

[17] K.M. Lee, Mining generalized fuzzy quantitative association rules with fuzzygeneralization hierarchies. *Joint 9th IFSA World Congress and 20th NAFIPS International Conference* (Vancouver, Canada, 2001) 2977-82.

[18] G. Raschia and N. Mouaddib, Saintetiq: A fuzzy set-based approach to database summarization, *Fuzzy Sets and Systems* 129(2) (2002) 137-62.

[19] R.A. Angryk and F.E. Petry, Mining multi-level associations with fuzzy hierarchies. *The 14th IEEE International Conference on Fuzzy Systems, FUZZ'05.* (2005) 785-90.

[20] U. Hierarchies, Database summarization using fuzzy isa hierarchies, *IEEE Transactions on Systems, Man, and Cybernetics-Part B* 27(1) (1997) 68-78.

[21] S. Tsumoto, Knowledge discovery in clinical databases and evaluation of discovered knowledge in outpatient clinic, *Information Sciences* 124(1) (2000) 125-37.

[22] M.L. Antonie and O.R. Zaiane, Mining Positive and Negative Association Rules: An Approach for Confined Rules, *PKDD*, (2004).

[23] A. Savasere, E. Omiecinski and S. Navathe, Mining for Strong Negative Associations in a Large Database of Customer Transactions, *Proceedings of the Fourteenth International Conference on Data Engineering*, (1998).

[24] X. Yuan, B.P. Buckles, Z. Yuan and J. Zhang, Mining Negative Association Rules, *Seventh International Symposium on Computers and Communications, 2002. (ISCC 2002)*, (2002).

[25] X. Wu, C. Zhang and S. Zhang, Efficient Mining of Both Positive and Negative Association Rules, *ACM Transactions on Information Systems (TOIS)*, Vol. 22 (3), (2004) 381-405.

[26] J. Han and Y. Fu, Mining multiple-level association rules in large databases, *IEEE Transactions on Knowledge and Data Engineering* 11(5) (1999) 798-805.

[27] B. Liu, W. Hsu and Y. Ma, Mining association rules with multiple minimum supports. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining* (San Diego, United States, 1999) 337-41.