Intelligent Information Management, 2012, 4, 217-224
http://dx.doi.org/10.4236/iim.2012.45032 Published Online September 2012 (http://www.SciRP.org/journal/iim)

A Study on Associated Rules and Fuzzy Partitions for Classification

Yeu-Shiang Huang, Jyi-Feng Yao
Department of Industrial and Information Management, National Cheng Kung University, Chinese Taipei
Email: yshuang@mail.ncku.edu.tw

Received May 12, 2012; revised June 22, 2012; accepted July 2, 2012

ABSTRACT

The amount of data for decision making has increased tremendously in the age of the digital economy. Decision makers who fail to proficiently manipulate the data produced may make incorrect decisions and therefore harm their business. Thus, the task of extracting and classifying the useful information efficiently and effectively from huge amounts of computational data is of special importance. In this paper, we consider that the attributes of data could be both crisp and fuzzy. By examining the suitable partial data, segments with different classes are formed, then a multithreaded computation is performed to generate crisp rules (if possible), and finally, the fuzzy partition technique is employed to deal with the fuzzy attributes for classification. The rules generated in classifying the overall data can be used to gain more knowledge from the data collected.

Keywords: Data Mining; Fuzzy Partition; Partial Classification; Association Rule; Knowledge Discovery

1. Introduction

Due to the rapid development of information technology, such as databases and networks, modern industry is able to gather and store large amounts of data easily. The collected data, which may include product sales, manufacturing records, supplier profiles, and customer information, etc., are utilized for transaction processes, information management, and decision support. The enormously increased amount of data may trigger a sensation of information overload, causing decision makers to be incapable of manipulating the data efficiently and effectively. This might result in incorrect decisions and therefore harm their business. Some customer-oriented companies have started to realize that it is necessary to pay more attention to customers and their preferences. The effective manipulation of large amounts of customer related information has been a critical success factor for corporation, since these useful (but often unorganized data), usualy stored in data warehouses, may contain information important for managerial decision making. Hence, the crucial task of discovering the deeply buried knowledge in data warehouses is of special importance and a source of strength for decision makers. The conceptual idea of knowledge discovery in database (*i.e.*, KDD) may be a good solution for dealing with such problems. It can be used to transform the data into valuable knowledge for supporting decision analyses and improving customer relationships. Recent research into knowledge discovery has been based on statistical methods, artificial intelligence, neural networks, genetic algorithms, and fuzzy inference, etc.

Ordinal classification approaches are generally focused on one specific type of data, especially numerical or nomial data. Such a limitation is impractical when the attributes of collected data are both numerical and nominal, and yet are both crisp and fuzzy. Association rules can be facilitated to classify desirable knowledge with associations by analyzing, matching, and combining the data, but such classification usually results in crisp rules. However, the possible fuzzy representations of some valuable attributes need to be critically concerned as well, and fuzzy inferences and fuzzy rule generations can thus play important roles in dealing with such problems.

In this paper, we make use of association rules to construct the classification knowledge, and at the same time, a multithread approach is employed to speed up the crisp rules generation process. Furthermore, in cases where the crisp rules generated are not sufficient for classification, the fuzzy rules would be utilized to cope with the fuzzy attributes and generate the ultimate rules for classification. This approach would significantly reduce the amount of computation required when compared with other approaches, such as neural networks or genetic algorithms. Therefore, the objective of this paper is to gain the classification knowledge that is buried in the data warehouse by using the approaches of association rules and fuzzy partitions in coping with both crisp and fuzzy data. In

Copyright © 2012 SciRes.

addition, we also study the effects of some important factors on the classification results, such as the number of classes and the supports. It is notable that the use of multithread technique would massively reduce the processing time required to achieve the desired results. Finally, this paper is organized as follows: Section 2 states the concepts of data association and classification in dealing with collected data; Section 3 shows the algorithms; Section 4 performs applications to validate the proposed schemes; and finally, Section 5 draws the concluding remarks.

2. Literature Review

Knowledge discovery has been a vital theme for corporations since 1991. The process of knowledge discovery includes data cleansing, integrating, selecting, transforming, mining, and interpreting [7,20]. KDD is a cyclic process to gain valuable knowledge, and data mining plays an important role in KDD [8]. Fayyad *et al.* [7] stated that data mining is "the nontrivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data." Data mining can extract information by the approaches of data association, classification, clustering, visualization, and template, etc., and the research areas of data mining include statistics, machine learning, data visualization, knowledge management, database techniques, and others [9]. Decision trees, neural networks, and genetic algorithms are often used to cope with the complicated problems of classification and prediction that arise in this process. This paper dealt with both crisp and fuzzy data by using the techniques of associations and classifications to gain effective knowledge for improved managerial decision making.

The major task of data association is to find the patterns of association among data, and then generate the association rules according to their predefined confidences and supports [1,5,10,22]. Han *et al.* [10] used the transaction data of supermarkets and found an interesting (and now famous) association rule that customers who bought diapers usually bought beers at the same time. This seems to be odd at first, since no one ever thought that diapers and beers could be allocated on shelves near each other in order to raise their sales, but it eventually turns out to be a great idea. Therefore, data association is a valuable technique to mine deeply buried, otherwise inconceivable information to be made available for managerial decision making. It is obvious that the whole set of association rules can be obtained by mining the database with the data association technique, but the crucial task will then be the pruning process for such large rule sets. The minimum coverage of these rules in the database is the threshold for retaining the useful association rules. The general procedure of data association consists of two steps [1]. First of all, to find the largest sets of items whose supports are greater than the preset value of support, and secondly, to generate the association rules by using the largest sets of items. Mannila *et al.* [18] and Park *et al.* [19] followed these basic steps and proposed their revised association procedure to improve the effectiveness of association mining.

Compared with the unsupervised method of data clustering, data classification is a supervised method that provides decision makers with the guidelines for decision analysis or outcomes prediction by identifying the rules of actual events [4,6,14]. Unlike data association, the results of data classification are not only interpretable and meaningful but also acceptable and understandable by decision makers. The techniques include induced decision trees, Bayesian classification, classification based on association rule, case-based reasoning, and fuzzy set approach, etc. Normally, the data are separated into two parts, *i.e.*, the training data and the test data. Learning and classifying are two major phases in data classification. During the learning process, the training data are managed to generate the classification rules, and then the classification process classifies the test data by using the classification rules generated from the learning process.

Han and Kamber [9] stated that the major concerns of data classification were predictive accuracy, speed, robustness, scalability, and interpretability. One crucial problem for data classification is that the amount of data is usually remarkably large, and it is very time consuming to perform the process of classification. Some research has used heuristic approaches to reduce the processing time. Bayardo [3] proposed a brute-force mining technique to generate classification rules with high confidence. Ali *et al.* [2] suggested a method of partial classification using association rules to resolve the problem of over computing. The major benefit of partial classification compared to traditional classification is that it provides a way to classify data more accurately and efficiently. Ali *et al.* [2] also stated that the main problems of inefficiency for classification might stem from too many attributes, missing values of attributes, not uniformly distributed classes, interdependent attributes, and too many training examples. However, these problems can be resolved by using the data association technique. Liu *et al.* [17] proposed an approach of class associative rules (CARs) and claimed that it could reduce over expanded amounts of classification rules, but the continuous types of data have to be preprocessed and transformed into discrete types, since CARs can only deal with discrete types of data. Li *et al.* [16] proposed another method of classification based on multiple class association rules (CMAR), which was improved from the CARs. They empirically showed that the average accurate rates for C4.5 [21], classification based on association (CBA), and CMAR were about 83.3%, 84.7%, and

Copyright © 2012 SciRes.

85.2%, respectively. It seems that classification by association can actually enhance the accuracy of classification.

Usually, the attributes of collected data are not all crisp. Some fuzzy attributes indeed exist in real world situations, such as customer satisfaction and customer loyalties, etc. The fuzziness from human cognitive processes can often be modeled by taking a fuzzy approach (*i.e.*, fuzzy numbers and membership functions) [11,12]. Kruse *et al.* [15] indicated that fuzzy systems have the deduction abilities to process semantic data by transforming them into mathematic structures. Ishibuchi *et al.* [13] proposed a classification method by using fuzzy rules to analyze the classification knowledge. He stated two basic methods for rule-based fuzzy systems: 1) voting by multiple fuzzy if-then rules in a single fuzzy rule-based classification system; 2) voting by multiple fuzzy rule-based classification systems.

The problem with fuzzy classification stems from either low accuracy because of the relatively wide fuzzy partitions or the large number of rules generated because of the relatively narrower fuzzy partitions. Certainly, neither of these conditions is ideal for decision makers. Yen [23] proposed three approaches to fuzzy partitions which are grid, scatter, and tree partitions, respectively. In this paper, the grid partition was used combined with the classification method proposed by Ishibuchi *et al.* [13], since they both produce more understandable results and are relatively straightforward to implement..

3. The Proposed Approach

The collected data for decision making usually has both crisp and fuzzy attributes. To cope with this problem, an approach was developed which combined the association rules and the fuzzy rules along with the multithread technique to integrate a classification system. Suppose that a database $\boldsymbol{D}$ has N transaction records $\{d_1,d_2, \cdots, d_N\}$, each transaction record has r crisp items and s fuzzy items, *i.e.*, $\boldsymbol{D} = \{d_1,d_2, \cdots, d_N\}$, $\boldsymbol{C} = \{C_1,C_2, \cdots,C_r\}$, and $\boldsymbol{F} = \{F_1,F_2, \cdots, F_s\}$, where $\boldsymbol{C}$ and $\boldsymbol{F}$ are the sets of crisp items and fuzzy items, respectively. Furthermore, we assume that the ith crisp term C_i has p_i different categories, and the jth fuzzy item F_j has q_j different categories, *i.e.*, $\boldsymbol{C}_i$:$(c_{ip1}, c_{ip2}, \cdots, c_{ipi})$, and $\boldsymbol{F}_j$:$(f_{jq1}, f_{jq2}, \cdots, f_{jqj})$. The possible number of different classes of the data set is assumed to be M, *i.e.*, $\boldsymbol{T}$:$(T_1,T_2, \cdots, T_M)$. Therefore, if we select n records from the database $\boldsymbol{D}$ to form the training data set $\boldsymbol{D}^T$ which is denoted by $\boldsymbol{D}^T = \{d^T{}_1,d^T{}_2, \cdots ,d^T{}_n\}$, the training data will be of the form as **Table 1**, where T_ID denotes the transaction identification.

The framework of the two-stage process of classification is shown in **Figure 1**, and the detail development of rule generation is shown in **Figure 2**.

The main tasks in generating the association rules are partitioning the training data by different classes, finding 1-ruleitem rules, generating association rules, using the multithread technique, creating the data for fuzzy classification, and generating fuzzy association rules, etc. We discuss these tasks in detail in the next few paragraphs.

For partitioning the training data into different classes, some SQL statements were simply implemented and the results are shown in **Table 2**.

Table 1. Training transaction data.

T_ID	C_1	C_2	⋯	C_r	F_1	F_2	⋯	F_s	Class
1	c_{11}	c_{21}	⋯	c_{r1}	f_{11}	f_{21}	⋯	f_{s1}	T_1
2	c_{12}	c_{22}	⋯	c_{r2}	f_{12}	f_{22}	…	f_{s2}	T_2
·	·	·	⋯	·	.	.	⋯	·	·
·	·	·	⋯	·	.	.	⋯	·	·
·	·	·	⋯	·	.	.	⋯	·	.
n	c_{1n}	$c_{2\text{n}}$	⋯	c_{rn}	f_{1n}	f_{2n}	⋯	f_{sn}	T_M

Table 2. The data table of Class Ti.

T_ID	C_1	C_2	⋯	C_r	F_1	F_2	⋯	F_s	Class
1	·	.	.	·	.	·	.	·	
2	·	.	.	·	.	·	.	·	
·	·	.	⋯	·	.	·	⋯	·	T_{i}
·	·	.	⋯	·	.	·	⋯	·	
·	·	.	⋯	·	.	·	⋯	·	
n_{i}	·	.	.	·	.	·	⋯	·	

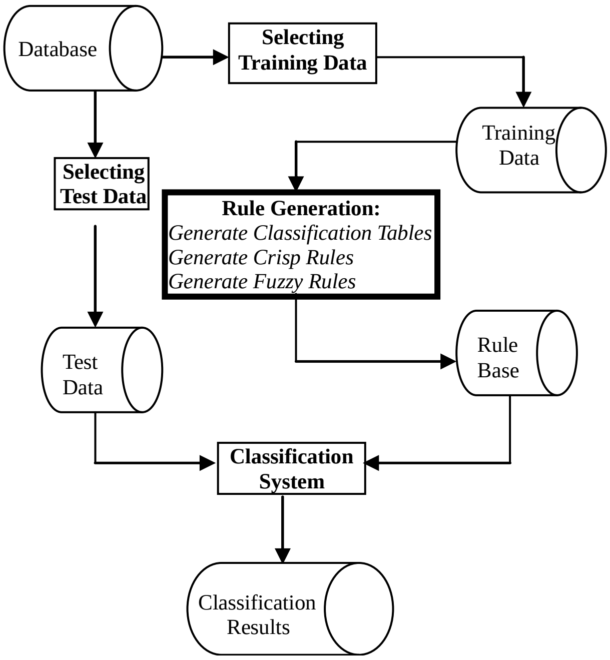

Figure 1. The two-stage process of classification.

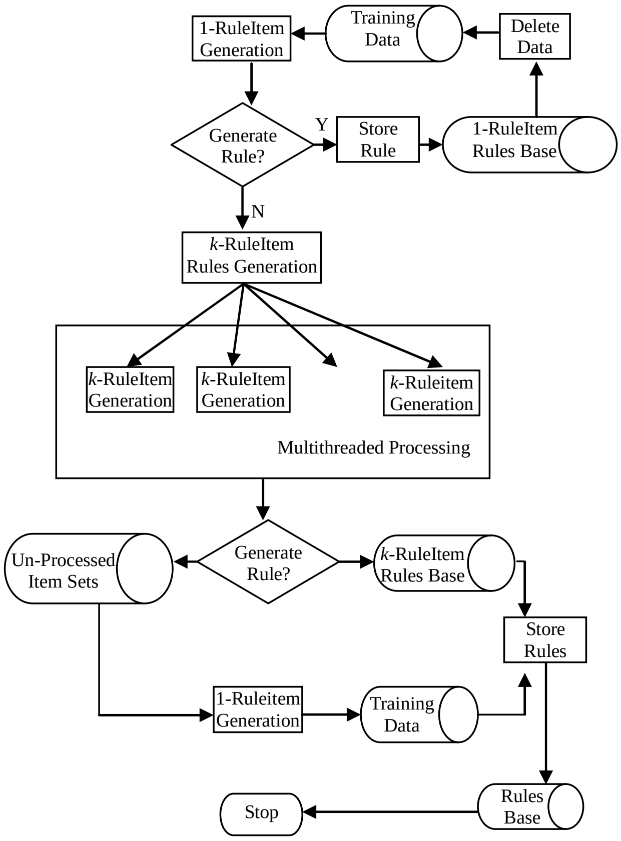

Figure 2. The rule generation process.

As shown in **Table 2**, each different class has its own table of transaction data, and each table can be used to generate the association rules for the same class. Note that

$$\sum_{i=1}^{M} n_i = n .$$

We first process the crisp rules. After the 1-rule item rules have been generated, the association rules can be found by the approach proposed by Han and Kamber [9]. Some revisions were made to fit our problem descriptions. The approach includes four parts, such as the main routine, the ruleitem sets generation routine, the deleting infrequent ruleitems routine, and the finding freqent 1-itemrules routine. This program is merely for one class, *i.e.*, T_i. To obtain more efficiency for classification, the multithreaded technique can be used to process the program mentioned previously in parallel for each of the *M* classes. After executing all the *M* processes in parallel, crisp rules can be generated for each class if they exist. Note that it is possible that the program for each class can result in similar association rules which have the same values for the crisp attributes but assign them to different classes. Such situations of the same rule but in a different class can happen because we only process the crisp attributes so far. The conflict rules have to be refined in order to obtain the rational rules after further examination by considering the fuzzy attributes. We store these conflict rules in the table of fuzzy processing rule (FPRT), and wait for the fuzzy classification.

For joining the two tables of $\boldsymbol{D}^T$ and FPRT to obtain the transaction data corresponding to the fuzzy items, the SQL statements are generated for fuzzy rules processing and an example table of results is shown in **Table 3**.

Table 3 shows that for certain number of data (e.g., n_l^*), the categories of their crisp items are all identical (e.g., ($c_1^*, c_2^*, \cdots, c_r^*$)), but they belong to different classes (e.g., $T_i, \cdots, T_j$). In such cases, further processing is necessary to be performed by considering the fuzzy items. We use the approach proposed by Ishibuchi *et al.* [13] for fuzzy classification. Triangular membership functions are employed to represent the four human judgment degrees of small, medium, medium large, and large, and a sample result of a fuzzy rule R_j can be given as follows:

$$R_j : IF\ F_1 = f_{1q_1} \text{ and } F_2 = F_{2q_2} \text{ and } F_s = f_{sq_s}$$
$$\text{then } ClassT_M \text{ with } CF_j$$

where CF_j denotes the grade of certainty for rule R_j [13]. Note that the algorithms used in this section are presented in Appendix.

4. Numerical Investigation

A set of data of customers' creditability for classification was simulated by the computer program. For simplicity without losing generality, six crisp and two fuzzy attributes are used, and one class attribute is utilized to classify the data. The sample data are shown in **Table 4**.

The crisp items are gender: {Female, Male}, annual income: {under 15K, 15K~20K, 20K~25K, 25K~30K, 30K~35K, over 35K}, loan: {None, 15K, 20K, 25K, 30 K}, saving: {under 15K, 15K~20K, 20K~25K, 25K~30K, 30K~35K, over 35K}, housing: {Yes, No}, and area: {East, West, South, North, Central}. The fuzzy items are customer loyalty: [0..1] and customer satisfaction: [0..1]. We assume that the customers have been preset to 21 classes, *i.e.*, from Class A to Class U, according to their credit records. From this data set, the possible combinations of customers from the crisp items are over 3,600, and the possible number of fuzzy rules are the partitions by the fuzzy items which is 4 × 4 = 16 for each conflict case mentioned in the previous section. 33,000 records of customer data are randomly simulated under the situation

Table 3. A processing table for fuzzy rules.

T_ID	C_1	C_2	⋯	C_r	F_1	F_2	⋯	F_s	Class
1	·	·	·	·	f_{11}	f_{21}	⋯	f_{s1}	
2	·	·	·	·	f_{12}	f_{22}	⋯	f_{s2}	T_i^*
·	c_1^*	c_2^*	⋯	c_r^*	·	·	⋯	·	·
·	·	·	⋯	·	·	·	⋯	·	·
·	·	·	⋯	·	·	·	⋯	·	T_j^*
n_l^*	·	·	·	·	$f_1 n_l^*$	$f_2 n_l^*$	⋯	$f_s n_l^*$	

Copyright © 2012 SciRes.

where 10 predefined crisp rules are included, and the amounts of data for each predefined crisp rule are between 300 to 500 and 1000 to 2000. The objective of putting predefined crisp rules in the data set is to verify that the proposed schema is actually capable of finding the hidden, predefined crisp rules. We set the minimal supports to be 0.2%, 0.4%, and 0.6%, to evaluate how the minimal support affects the accuracy in identifying the predefined crisp rules. We set the minimal confidence to 60%. We also took 25%, 50%, and 75% of the data set as the training data to study the different effects of partitioning.

The final results after running the classification process would be two types of rules. One is the crisp rule, since the rule can be generated merely by the crisp items, and no further process is needed. An example is given by *IF* "sex = F *AND* annual income = 25K ~ 30K *AND* loan = None *AND* saving = Under 15K *AND* housing = No *AND* area = North" *THEN* "class = A", and the other is the fuzzy rule, which consists of crisp and fuzzy items, since the rule can not be decided alone by the crisp items, and the fuzzy items need to be taken into consideration. An example is also given by *IF* "sex = F *AND* annual income = 25K ~ 30K *AND* loan = None *AND* saving = Under 15K *AND* housing = No *AND* area = North" *AND* *IF* "customer loyalty = Small and customer satisfactory = Large" *THEN* "class = A" *WITH CF* = 0.7.

Table 5 shows the classification results for the different minimal supports.

Note that the accuracy rate denotes the percentage that the predefined rules are correctly identified by the generated rules. From **Table 6** (the gray area) we can see that the proposed approach can accurately identify the predefined crisp rules for most of the cases. However, increasing the minimal support seems to reduce the number of correct rules generated and the accuracy rates. This is obviously true, since the minimal support is the threshold for rule generations [10].

Table 4. Simulated customer transaction data.

Gender	Annual income	Loan	Saving	Housing	Area	Customer loyalty	Customer satisfaction	Class
F	25K ~ 30K	None	Under 15K	No	North	0.31	0.75	A
F	Over 35K	20K	25K ~ 30K	Yes	East	0.52	0.84	D
M	20K ~ 25K	15K	20K ~ 25K	Yes	East	0.23	0.76	B
F	25K ~ 30K	20K	25K ~ 30K	No	South	0.78	0.69	A
M	30K ~ 35K	25K	30K ~ 35K	Yes	West	0.65	0.54	E
-	-	-	-	-	-	-	-	-
-	-	-	-	-	-	-	-	-

Table 5. The results of classification for crisp rules.

Percentage of training data	The number of data	Minimal support	The number of rule generated	The number of predefined rules	Accuracy rate
25%	8250	0.2%	11	10	100%
		0.4%	10	10	100%
		0.6%	10	10	100%
50%	16500	0.2%	12	10	100%
		0.4%	10	10	100%
		0.6%	9	10	90%
75%	24750	0.2%	12	10	100%
		0.4%	10	10	90%
		0.6%	9	10	90%

Table 6. Accuracy 6 of fuzzy classification 6.

Number of class Training data (%)	20			30			50		
	2 × 2	5 × 5	10 × 10	2 × 2	5 × 5	10 × 10	2 × 2	5 × 5	10 × 10
25%	87%	60%	48%	76%	41%	28%	56%	43%	24%
50%	89%	69%	56%	79%	48%	33%	59%	45%	30%
75%	93%	73%	59%	82%	50%	36%	63%	56%	35%

For the rules that cannot be decided merely by the crisp items, the further process for fuzzy classification was then performed. We preset the threshold of *CF* (*i.e.*, the grade of certainty) to be 25%. If, for example, after running the training process, a fuzzy rule claims that some customer data can be classified as Class A with CF_A = 30%, as Class B with CF_B = 25%, as Class C with CF_C = 20%, as Class D with CF_D = 15%, and as Class E with CF_E = 10%, respectively, then it will be treated as correct if the test data classifies the certain customer data as A or B, otherwise (e.g., classified as C, D, or E) it will be treated as incorrect. **Table 6** shows the results of fuzzy classification.

As shown in **Table 6** (the gray area), for our simulated data which has 50 classes and 10 × 10 fuzzy partitions, the accuracy rates are not very encouraging. The reason is that there are too many classes to be classified and also too many categories of the fuzzy attributes to be partitioned, *i.e.*, the more the fuzzy partitions the less certainty a pattern may belong to a class, so the number of data in each partition is therefore very small which may cause the *CF* to be close to 1. It means that every customer data record may produce a single rule, thus, the chance to find another identical record in the test data is very small and as a result the accuracy rate is also small. In such a case, in order to improve the classification accuracy, a large amount of data is required. We performed several experiments by decreasing the number of classes and the partitions of fuzzy attributes, and the results are also shown in **Table 6** (the gray area). As can be seen in the figure, the accuracy rate increases when the numbers of classes and the fuzzy partitions both decrease. The percentage of training data used is also an affecting factor. This is understandable, since when we use more data to generate classification rules, the rules generated will be more statistically accurate.

5. Conclusion

This paper deals with the hybrid systems which contain both the crisp and fuzzy attributes. The associations with crisp items are used firstly to generate the crisp rules, and the multithread technique is also employed at this stage. Then the further process is required to use fuzzy items to classify data if some crisp rules are not successfully generated. This combined approach can improve the task of classification more efficiently and effectively. However, the problems of a large number of classes and a large number of categories for fuzzy items are the issues of importance which might need to be further investigated.

REFERENCES

[1] R. Agrawal, T. Imielinski and A. Swami, “Mining Association Rules between Sets of Items in Large Databases,” *Proceeding of the ACM SIGMOD Conference on Management of Data*, Washington DC, 26-28 May 1993, pp. 207-216.

[2] K. Ali, S. Manganaris and R. Srikant, “Partial Classification Using Association Rules,” *Proceeding of the 3rd International Conference on Knowledge Discovery and Data Mining*, (*KDD*-97), Newport Beach, 14-17 August 1997, pp. 115-118.

[3] R. J. Bayardo, “Brute-Force Mining of High Confidence Classification Rules,” *Proceeding of the 3rd International Conference on Knowledge Discovery and Data Mining*, (KDD-97), Newport Beach, 14-17 August 1997, pp. 123-126.

[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*, Vol. 8, No. 6, 1996, pp. 866-883. doi:10.1109/69.553155

[5] F. Coenen, P. Leng and L. Zhang, “Threshold Tuning for Improved Classification Association Rule Mining,” *Advances in Knowledge Discovery and Data Mining*, Vol. 3518, 2005, pp. 303-305. doi:10.1007/11430919_27

[6] A. Evfimievski, R. Srikant, R. Agrawal and J. Gehrke, “Privacy Preserving Mining of Association Rules,” *Information Systems*, Vol. 29, No. 4, 2004, pp. 343-364. doi:10.1016/j.is.2003.09.001

[7] U. M. Fayyad, G. Piatetsky-Shapiro and P. Smyth, “From Data Mining to Knowledge Discovery: An Overview,” In: U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth and R. Uthurusamy, Eds., *Advances in Knowledge Discovery and Data Mining*, AAAI Press, Menlo Park, 1996, pp. 1-34.

[8] A. A. Freitas, “A Survey of Evolutionary Algorithms for Data Mining and Knowledge Discovery,” *Advances in Evolutionary Computing—Natural Computing Series*, *Part* II, Springer-Verlag, New York, 2003, pp. 819-845.

[9] J. Han and M. Kamber, “Data Mining Concepts and Techniques,” Morgan Kaufmann, San Francisco, 2001.

[10] J. Han, J. Pei and Y. Yin, “Mining Frequent Patterns without Candidate Generation,” Computing Science Technical Report, TR-99-12, Simon Fraser University, Burnaby, 1999.

[11] Y.-C. Hu, R.-S. Chen and G.-H. Tzeng, “Finding Fuzzy Classification Rules Using Data Mining Techniques,” *Pattern Recognition Letters*, Vol. 24, No. 3, 2003, pp. 509-519. doi:10.1016/S0167-8655(02)00273-8

[12] H. Ishibuchi and T. Yamamoto, “Fuzzy Rule Selection by Multi-Objective Genetic Local Search Algorithms and Rule Evaluation Measures in Data Mining,” *Fuzzy Sets and Systems*, Vol. 141, No. 1, 2004, pp. 59-88. doi:10.1016/S0165-0114(03)00114-3

[13] H. Ishibuchi, T. Nakashima and T. Morisawa, “Voting in Fuzzy Rule-Based Systems for Pattern Classification Problems,” *Fuzzy Sets and Systems*, Vol. 103, No. 2, 1999, pp. 223-238. doi:10.1016/S0165-0114(98)00223-1

[14] M. Kantarcioglu and C. Clifton, “Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data,” *IEEE Transactions on Knowledge and*

Data Engineering, Vol. 16, No. 9, 2004, pp. 1026-1037. doi:10.1109/TKDE.2004.45

[15] R. Kruse, J. Gebhardt and F. Klawonn, "Foundations of Fuzzy Data," John Wiley & Sons, Chichester, 1994.

[16] W. Li, J. Han and J. Pei, "CMAR: Accurate and Efficient Classification Based on Multiple Class-Association Rules," *Proceeding of* 2001 *International Conference on Data Mining* (ICDM'01), San Jose, 2001.

[17] B. Liu, W. Hsu and Y. M. Ma, "Integrating Classification and Association Rule Mining," *Proceeding of the* 4*th International Conference on Knowledge Discovery and Data Mining* (KDD-98), New York, 27-31 August 1998, pp. 80-86.

[18] H. Mannila, H. Toivonen and I. Verkamo, "Efficient Algorithms for Discovering Association Rules," *AAAI Workshop on Knowledge Discovery in Databases*, Seattle, 1994.

[19] J. S. Park, M. Chen and P. S. Yu, "An Effective Hash Based Algorithm for Mining Association Rules," *Proceeding of the ACM SIGMOD International Conference on Management of Data*, New York, 1995, pp. 175-186.

[20] G. Piatetsky-Shapiro and W. J. Frawley, "Knowledge Discovery in Databases: An Overview," MIT Press, Cambridge, 1991.

[21] J. R. Quinlan, "C4.5: Programs for Machine Learning," Morgan Kaufmann, California, 1993.

[22] F. Thabtah, "A Review of Associative Classification Mining," *The Knowledge Engineering Review Archive*, Vol. 22, No. 1, 2007, pp. 37-65. doi:10.1017/S0269888907001026

[23] J. Yen, "Fuzzy Logic—A Modern Perspective," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 11, No. 1, 1999, pp. 153-165. doi:10.1109/69.755624

Appendix

A.1 SQL Statements for Partitioning the Training Data

```
Insert into Table_i
select * from $\boldsymbol{D}^T$
where Class = "$T_i$"
```

A.2 The Generation of 1-RuleItem Rules

```
for i = 1 to r ;
{m = $C_i$.number_of_attribute; for j = 1 to m;
{ Count $C_{ij}$; Count $C_{ij}$.Class
when $C_{ij}$.Count = $C_{ij}$.Class.Count
Ruleij = <$C_{ij}$, Class>;
delete $d^T_j$ (where $C_{ij}$.Value=$C_{ij}$)}}
```

A.3 The Generation of *k*-Ruleitem Rules

```
Generate_itemrules_of_Class$T_i$ ()
$L_1$ = find_frequent_1-itemrules ($\boldsymbol{D}^T$);
for(k = 2;$L_{k-1}$! = $\emptyset$;k + + ) {$C_k$ = apriori_gen($L_{k-1}$, minsup); for each transaction $t \in D^T$ {$C_t$ = subset ($C_k$,t); for each candidate $c \in C_t$ c.count++;}}
return L= $\cup_k L_k$;
appriori_gen($L_{k-1}$,minsup)
for each itemset $l_1 \in L_{k-1}$
for each itemset $l_2 \in L_{k-1}$
if ($l_1$ [1] = $l_2$ [1]^ $l_1$ [2] = $l_2$ [2]^ ⋯⋯ $l_1$ [k-2] = $l_2$[k-2]^ $l_1$[k-1] = $l_2$[k-1]) then {c = $l_1 \rhd\lhd l_2$; if has infrequent_subset(c,$L_{k-1}$) then delete c; else add c to $C_k$; }}
infrequent_subset(c,$L_{k-1}$)
for each (k-1)-subset s of c
if s $\notin L_{k-1}$ then return TRUE; return FALSE;
find_frequent_1-itemrules($\boldsymbol{D}^T$)
for i = 1 to r
{ for j = 1 to m { if $C_i$. $c_{ij} \notin L_1$;
then add [$C_i$.$c_{ij}$] to $L_1$;
else
 j++;}
i++;}
```

A.4 The Algorithm of Multithreaded Computing for *M* Classes

```
Multithread( )
{ ThreadStart Generate1 = new ThreadStart (GenerateRule1);
ThreadStart Generate2 = new ThreadStart (GenerateRule2);
 …
 ThreadStart GenerateM= new ThreadStart(GenerateRuleM);
 Thread g1= new Thread (Generate1);
 Thread g2= new Thread (Generate2);
 …
Thread gM=new Thread (GenerateM);
g1.Start;
g2.Start;
…
gM.Start;}
```

A.5 The Generation of FPRT

```
select Table_1.*
from (Table_1 inner join Table_2 on (Table_1.* = Table_2.*) inner join Table_3 on (Table_2.* = Table_3.*) ……inner join Table_M on (Table_M-1.* = Table_M.*)
insert into FPRT;
```

A.6 SQL Statements to Join Two Data Tables

```
insert into FUZZY_ITEMSET_TABLE
select $\boldsymbol{D}^T$.$C_1$ to $\boldsymbol{D}^T$.$C_r$
from ($\boldsymbol{D}^T$ inner join FPRT on ($\boldsymbol{D}^T$. $C_1$ to $C_r$ = FPRTable.$C_1$ to $C_r$)
```

A.7 SQL Statements for Finding Fuzzy RuleItmes and Corresponding Classes

```
select [$F_1$], [$F_2$], ⋯,[$F_s$],[Class T]
from FUZZY_ITEMSET_TABLE;
select [$F_1$],[$F_2$], ⋯ [$F_s$], [Class T]
from FUZZY_ITEMSET_TABLE
where $F_1$ = "$f_{1q1}$"and $F_2$ = "$f_{2q2}$" and⋯and $F_s$ = "$f_{sqs}$".
```

Copyright © 2012 SciRes.