Peculiarity Oriented Multi-database Mining

Ning Zhong[1], Y.Y. Yao[2], and Setsuo Ohsuga[3]

[1] Dept. of Computer Science and Sys. Eng., Yamaguchi University
[2] Dept. of Computer Science, University of Regina
[3] Dept. of Information and Computer Science, Waseda University

Abstract. The paper proposes a way of mining *peculiarity rules* from multiply statistical and transaction databases. We introduce the *peculiarity rules* as a new type of association rules, which can be discovered from a relatively small number of the *peculiar* data by searching the relevance among the peculiar data. We argue that the peculiarity rules represent a typically unexpected, interesting regularity hidden in statistical and transaction databases. We describe how to mine the peculiarity rules in the multi-database environment and how to use the RVER (Reverse Variant Entity-Relationship) model to represent the result of multi-database mining. Our approach is based on the database reverse engineering methodology and granular computing techniques.

Keywords: Multi-Database Mining, Peculiarity Oriented, Relevance, Database Reverse Engineering, Granular Computing (GrC).

1 Introduction

Recently, it has been recognized in the KDD (Knowledge Discovery and Data Mining) community that *multi-database mining* is an important research topic [3, 14, 19]. So far most of the KDD methods that have been developed are on the single universal relation level. Although theoretically, any multi-relational database can be transformed into a single universal relation, practically this can lead to many issues such as universal relations of unmanageable sizes, infiltration of uninteresting attributes, losing of useful relation names, unnecessary join operation, and inconveniences for distributed processing. In particular, some concepts, regularities, causal relationships, and rules cannot be discovered if we just search a single database since the knowledge hides in multiply databases basically.

Multi-database mining involves many related topics including interestingness checking, relevance, database reverse engineering, granular computing, and distributed data mining. Liu et al. proposed an interesting method for relevance measure and an efficient implementation for identifying relevant databases as the first step for multi-database mining [10]. Ribeiro et al. described a way for extending the INLEN system for multi-database mining by the incorporation of primary and foreign keys as well as the development and processing of knowledge segments [11]. Wrobel extended the concept of foreign keys into foreign links because multi-database mining is also interested in getting to non-key attributes

J.M. Żytkow and J. Rauch (Eds.): PKDD'99, LNAI 1704, pp. 136–146, 1999.

[14]. Aronis et al. introduced a system called WoRLD that uses spreading activation to enable inductive learning from multiple tables in multiple databases spread across the network [3].

Database reverse engineering is a research topic that is closely related to multi-database mining. The objective of database reverse engineering is to obtain the domain semantics of legacy databases in order to provide meaning of their executable schemas' structure [6]. Although database reverse engineering has been investigated recently, it was not researched in the context of multi-database mining. In this paper we take a unified view of multi-database mining and database reverse engineering. We use the RVER (Reverse Variant Entity-Relationship) model to represent the result of multi-database mining. The RVER model can be regarded as a variant of semantic networks that are a kind of well-known method for knowledge representation. From this point of view, multi-database mining can be regarded as a kind of database reverse engineering.

A challenge in multi-database mining is semantic heterogeneity among multiple databases since no explicit foreign key relationships exist among them usually. Hence, the key issue is how to find/create the relevance among different databases. In our methodology, we use *granular computing* techniques based on semantics, approximation, and abstraction [7, 18]. Granular computing techniques provide a useful tool to find/create the relevance among different databases by changing information granularity.

In this paper, we propose a way of mining *peculiarity* rules from multiply statistical and transaction databases, which is based on the database reverse engineering methodology and granular computing techniques.

2 Peculiarity Rules and Peculiar Data

In this section, we first define *peculiarity rules* as a new type of association rules and then describe a way of finding peculiarity rules.

2.1 Association Rules vs. Peculiarity Rules

Association rules are an important class of regularity hidden in transaction databases [1, 2]. The intuitive meaning of such a rule is that transactions of the database which contain X tend to contain Y. So far, two categories of the association rules, the *general rule* and the *exception rule*, have been investigated [13]. A *general rule* is a description of a regularity for numerous objects and represents the well-known fact with common sense, while an *exception rule* is for a relatively small number of objects and represents exceptions to the well-known fact. Usually, the exception rule should be associated with a general rule as a set of rule pairs. For example, the rule "using a seat belt is risky for a child" which represents exceptions to the general rule with common sense "using a seat belt is safe".

The *peculiarity rules* introduced in this paper can be regarded as a new type of association rules for a different purpose. A peculiarity rule is discovered from the *peculiar* data by searching the relevance among the *peculiar* data. Roughly speaking, a data is *peculiar* if it represents a peculiar case described by a relatively small number of objects and is very different from other objects in a data set. Although it looks like the exception rule from the viewpoint of describing a relatively small number of objects, the peculiarity rule represents the well-known fact with common sense, which is a feature of the general rule.

We argue that the *peculiarity rules* are a typical regularity hidden in statistical and transaction databases. Sometimes, the general rules that represent the well-known fact with common sense cannot be found from numerous statistical or transaction data, or although they can be found, the rules may be uninteresting ones to the user since data are rarely specially collected/stored in a database for the purpose of mining knowledge in most organizations. Hence, the evaluation of interestingness (including surprisingness, unexpectedness, peculiarity, usefulness, novelty) should be done before and/or after knowledge discovery [5, 9, 12]. In particular, unexpected (common sense) relationships/rules may be hidden a relatively small number of data. Thus, we may focus some interesting data (the peculiar data), and then we find more novel and interesting rules (peculiarity rules) from the data.

For example, the following rules are the peculiarity ones that can be discovered from a relation called *Japan-Geography* (see Table 1) in a *Japan-Survey* database:

$rule_1 : ArableLand(large) \; \& \; Forest(large) \rightarrow PopulationDensity(low).$

$rule_2 : ArableLand(small) \; \& \; Forest(small) \rightarrow PopulationDensity(high).$

Table 1. Japan-Geography

Region	Area	Population	PopulationDensity	PeasantFamilyN	ArableLand	Forest	...
Hokkaido	82410.58	5656	**67.8**	93	**1209**	**5355**	...
Aomori	9605.45	1506	156.8	87	169	623	...
...	...	...	...	...	...	...	...
Tiba	5155.64	5673	1100.3	116	148	168	...
Tokyo	2183.42	11610	**5317.2**	21	**12**	**80**	...
...	...	...	...	...	...	...	...
Osaka	1886.49	8549	**4531.6**	39	**18**	**59**	...
...	...	...	...	...	...	...	...

In order to discover the rules, we first need to search the peculiar data in the relation *Japanese-Geography*. From Table 1, we can see that the values of the attributes *ArableLand* and *Forest* for Hokkaido (i.e. 1209 Kha and 5355 Kha) and for Tokyo and Osaka (i.e. 12 Kha, 18 Kha, and 80 Kha, 59 Kha) are very different from other values in the attributes. Hence, the values are regarded as the peculiar data. Furthermore, $rule_1$ and $rule_2$ are generated by searching the relevance among the peculiar data. Note that we use the qualitative representation for the quantitative values in the above rules. The transformation of quantitative to qualitative values can be done by using the following background knowledge on information granularity:

Basic granules:
bg_1 = {high, low}; bg_2 = {large, small}; bg_3 = {many, little};
bg_4 = {far, close}; bg_5 = {long, short}; ....

Specific granules:
biggest-cities = {Tokyo, Osaka}; kanto-area = {Tokyo, Tiba, Saitama, ...};
kansei-area = {Osaka, Kyoto, Nara, ...}; ....

That is, *ArableLand = 1209, Forest = 5355* and *PopulationDensity = 67.8* for Hokkaido are replaced by the granules, "large" and "low", respectively. Furthermore, Tokyo and Osaka are regarded as a neighborhood (i.e. the biggest cities in Japan). Hence, $rule_2$ is generated by using the peculiar data for both Tokyo and Osaka as well as their granules (i.e. "small" for *ArableLand* and *Forest,* and "high" for *PopulationDensity*).

2.2 Finding the Peculiar Data

There are many ways of finding the peculiar data. In this section, we describe an attribute-oriented method.

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a data set related to an attribute in a relation, and n is the number of different values in an attribute. The peculiarity of x_i can be evaluated by the *Peculiarity Factor*, $PF(x_i)$,

$$PF(x_i) = \sum_{j=1}^{n} \sqrt{N(x_i, x_j)}. \tag{1}$$

It evaluates whether x_i occurs relatively small number and is very different from other data x_j by calculating the sum of the square root of the conceptual distance between x_i and x_j. The reason why the square root is used in Eq. (1) is that we prefer to evaluate more near distances for relatively large number of data so that the peculiar data can be found from relatively small number of data.

Major merits of the method are

- It can handle both the continuous and symbolic attributes based on a unified semantic interpretation;
- Background knowledge represented by binary neighborhoods can be used to evaluate the peculiarity if such background knowledge is provided by a user.

If X is a data set of a continuous attribute and no background knowledge is available, in Eq. (1),

$$N(x_i, x_j) = |x_i - x_j|. \tag{2}$$

Table 2 shows an example for the calculation. On the other hand, if X is a data set of a symbolic attribute and/or the background knowledge for representing the conceptual distances between x_i and x_j is provided by a user, the peculiarity factor is calculated by the conceptual distances, $N(x_i, x_j)$. Table 3 shows an example in which the binary neighborhoods shown in Table 4 are used as the background knowledge for representing the conceptual distances of different type of restaurants [7, 15]. However, all the conceptual distances are 1, as default, if background knowledge is not available.

Table 2. An example of the peculiarity factor for a continue attribute

Region	**ArableLand**		**PF**
Hokkaido	**1209**		*134.1*
Tokyo	12		60.9
Osaka	18	=⇒	60.3
Yamaguchi	162		60.5
Okinawa	147		59.4

Table 3. An example of the peculiarity factor for a symbolic attribute

Restaurant	**Type**		**PF**
Wendy	American		2.2
Le Chef	French		2.6
Great Wall	Chinese	=⇒	1.6
Kiku	Japanese		1.6
South Sea	Chinese		1.6

Table 4. The binary neighborhoods for a symbolic attribute

Type	**Type**	**N**
Chinese	Japanese	1
Chinese	American	3
Chinese	French	4
American	French	2
American	Japanese	3
French	Japanese	3

After the evaluation for the peculiarity, the peculiar data are elicited by using a threshold value,

$$threshold = mean\ of\ PF(x_i) + \alpha \times variance\ of\ PF(x_i) \tag{3}$$

where α can be specified by a user. That is, if $PF(x_i)$ is over the threshold value, x_i is a peculiar data.

Based on the preparation stated above, the process of finding the peculiar data can be outlined as follows:

Step 1. Calculate the peculiarity factor $PF(x_i)$ in Eq. (1) for all values in a data set (i.e. an attribute).

Step 2. Calculate the threshold value in Eq. (3) based on the peculiarity factor obtained in *Step* 1.

Step 3. Select the data that is over the threshold value as the peculiar data.

Step 4. If current peculiarity level is enough, then goto *Step* 6.

Step 5. Remove the peculiar data from the data set and thus, we get a new data set. Then go back to *Step* 1.

Step 6. Change the granularity of the peculiar data by using background knowledge on information granularity if the background knowledge is available.

Furthermore, the process can be done in a parallel-distributed mode for multiple attributes, relations and databases since this is an attribute-oriented finding method.

2.3 Relevance among the Peculiar Data

A peculiarity rule is discovered from the peculiar data by searching the relevance among the peculiar data. Let $X(x)$ and $Y(y)$ be the peculiar data found in two attributes X and Y respectively. We deal with the following two cases:

- If the $X(x)$ and $Y(y)$ are found in a relation, the relevance between $X(x)$ and $Y(y)$ is evaluated in the following equation:

$$R_1 = P_1(X(x)|Y(y))P_2(Y(y)|X(x)). \tag{4}$$

  That is, the larger the product of the probabilities of P_1 and P_2, the stronger the relevance between $X(x)$ and $Y(y)$.
- If the $X(x)$ and $Y(y)$ are found in two different relations, we need to use a value (or its granule) in a key (or foreign key/link) as the relevance factor, $K(k)$, to find the relevance between $X(x)$ and $Y(y)$. Thus, the relevance between $X(x)$ and $Y(y)$ is evaluated in the following equation:

$$R_2 = P_1(K(k)|X(x))P_2(K(k)|Y(y)). \tag{5}$$

Furthermore, Eq. (4) and Eq. (5) are suitable for handling more than two peculiar data found in more than two attributes if $X(x)$ (or $Y(y)$) is a granule of the peculiar data.

3 Mining Peculiarity Rules in Multi-Database

Building on the preparatory in Section 2, this section describes a methodology of mining peculiarity rules in multi-database.

3.1 Multi-Database Mining in Different Levels

Generally speaking, the task of multi-database mining can be divided into two levels:

1. Mining from multiple relations in a database.
2. Mining from multiple databases.

First, we need to extend the concept of foreign keys into foreign links because we are also interested in getting to non-key attributes for data mining from multiple relations in a database. A major work is to find the peculiar data in multiple relations for a given discovery task while foreign link relationships exist. In other words, our task is to select n relations, which contain the peculiar data, among m relations ($m \geq n$) with foreign links.

We again use the *Japan-Survey* database as an example. There are many relations (tables) in this database such as *Japan-Geography, Economy, Alcoholic-Sales, Crops, Livestock-Poultry, Forestry, Industry*, and so on. Table 5 and Table 6 show two of them as examples (Table 1 is another one (Japan-Geography)). The method for selecting n relations among m relations can be briefly described as follows:

Table 5. Economy

Region	PrimaryInd	SecondaryInd	TertiaryInd	...
Hokkaido	**9057**	34697	96853	...
Aomori	2597	6693	22722	...
...	...	...	...	...
Tiba	3389	44257	76277	...
Tokyo	839	187481	**484294**	...
...	...	...	...	...
Osaka	397	99482	**209492**	...
...	...	...	...	...

Table 6. Alcoholic-Sales

Region	Sake	Beer	...
Hokkaido	42560	257125	...
Aomori	18527	60425	...
...	...	...	...
Tiba	47753	205168	...
Tokyo	**150767**	838581	...
...	...	...	...
Osaka	**100080**	577790	...
...	...	...	...

Step 1. Focus on a relation as the main table and find the peculiar data from this table. Then elicit the peculiarity rules from the peculiar data by using the methods stated in Section 2.2 and 2.3.

For example, if we select the relation called *Japan-Geography* shown in Table 1 as the main table, $rule_1$ and $rule_2$ stated in Section 2.1 are a result for the step.

Step 2. Find the value(s) of the focused key corresponding to the mined peculiarity rule in *Step* 1 and change its granularity of the value(s) of the focused key if the background knowledge on information granularity is available.

For example, “Tokyo” and “Osaka” that are the values of the key attribute *region* can be changed into a granule, “biggest cities”.

Step 3. Find the peculiar data in the other relations (or databases) corresponding to the value (or its granule) of the focused key.

Step 4. Select n relations that contain the peculiar data, among m relations ($m \geq n$). In other words, we just select the relations that contain the peculiar data that are relevant to the peculiarity rules mined from the main table.

Here we need to find the related relations by using foreign keys (or foreign links). For example, since the (foreign) key attribute is *Region* for the relations in the *Japan-Survey* database, and the value in the key, *Region = Hokkaido*, which is related to the mined $rule_1$, we search the peculiar data in other relations that are relevant to the mined $rule_1$ by using *Region = Hokkaido* as a relevance factor. The basic method for searching the peculiar data is similar to the one stated in Section 2.2. However, we just check the peculiarity of the data that are relevant to the value (or its granule) of the focused key in the relations. Furthermore, selecting n relations among m relations can be done in a parallel-distributed cooperative mode.

Let "|" denote a relevance among the *peculiar* data (but not a rule currently, and can be used to induce rules as to be stated in Section 3.2). Thus, we can see that the peculiar data are found in the relations, *Crops, Livestock-Poultry, Forestry, Economy*, corresponding to the value of the focused key, *Region = Hokkaido*:

In the relation, *Crops*,
Region(Hokkaido) | (*WheatOutput(high)* & *RiceOutput(high)*).
In the relation, *Livestock-Poultry*,
Region(Hokkaido) | (*MilchCow(many)* & *MeatBull(many)* & *MilkOutput(many)* & *Horse(many)*).
In the relation, *Forestry*,
Region(Hokkaido) | (*TotalOutput(high)* & *SourceOutput(high)*).
In the relation, *Economy*,
Region(Hokkaido) | *PrimaryIndustry(high)*.

Hence the relations, *Crops, Livestock-Poultry, Forestry, Economy* are selected. On the other hand, the peculiar data are also found in the relations, *Alcoholic-Sales* and *Economy*, corresponding to the value of the focused key, *Region = biggest-cities*:

In the relation, *Alcoholic-Sales*,
Region(biggest-cities) | (*Sake-sales(high)* & *RiceOutput(high)*).
In the relation, *Economy*,
Region(biggest-cities) | *TertiaryIndustry(high)*.

Furthermore, the methodology stated above can be extended for mining from multiple databases. For example, if we found that the turnover was a marked drop in some day from a supermarket transaction database, maybe we cannot understand why. However, if we search a weather database, we can find that there was a violent typhoon this day in which the turnover of the supermarket was a marked drop. Hence, we can discover the reason why the turnover was a marked drop.

A challenge in multi-database mining is semantic heterogeneity among multiple databases since no explicit foreign key relationships exist among them usually. Hence, the key issue is how to find/create the relevance among different databases. In our methodology, we use *granular computing* techniques based on semantics, approximation, and abstraction for solving the issue [7, 18].

3.2 Representation and Re-learning

We use the RVER (Reverse Variant Entity-Relationship) model to represent the peculiar data and the conceptual relationships among the peculiar data discovered from multiply relations (databases). Figure 1 shows the general framework of the RVER model. The RVER model can be regarded as a variant of semantic networks that are a kind of well-known method for knowledge representation. From this point of view, multi-database mining can be regarded as a kind of database reverse engineering. Figure 2 shows a result mined from the Japan-Survey database; Figure 3 shows the result mined from two databases on the supermarkets at Yamaguchi prefecture and the weather of Japan. The point of which the RVER model is different from an ordinary ER model is that we just represent the attributes that are relevant to the peculiar data and the related peculiar data (or their granules) in the RVER model. Thus, the RVER model provides all interesting information that is relevant to some focusing (e.g. *Region = Hokkaido* and *Region = biggest-cities* in the Japan-Geography database) for learning the advanced rules among multiple relations (databases).

Re-learning means learning the advanced rules (e.g., if-then rules and first-order rules) from the RVER model. For example, the following rules can be learned from the RVER models shown in Figure 2 and Figure 3:

$$rule_3 : ArableLand(large)\ \&\ Forest(large) \rightarrow PrimaryIndustry(high).$$

$$rule_4 : Weather(typhoon) \rightarrow Turnover(very\text{-}low).$$

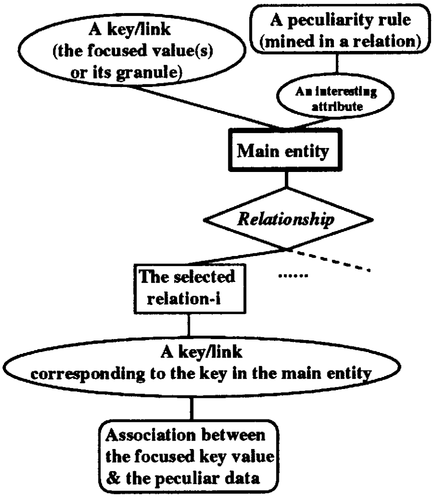
A key/link (the focused value(s) or its granule)
A peculiarity rule (mined in a relation)
An interesting attribute
Main entity
Relationship
The selected relation-i
......
A key/link corresponding to the key in the main entity
Association between the focused key value & the peculiar data

Fig. 1. The RVER model

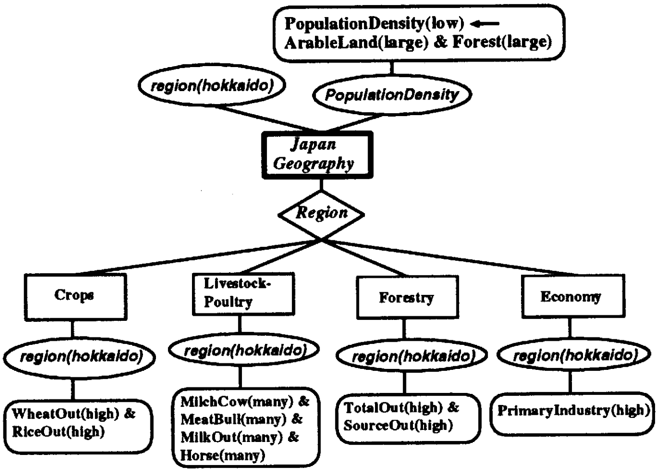

Fig. 2. The RVER model related to *Region* = *Hokkaido*

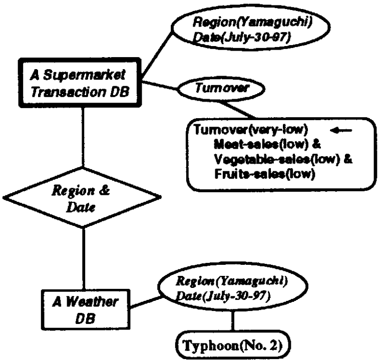

Fig. 3. The RVER model mined from two databases

4 Conclusion

We presented a way of mining peculiarity rules from multiply statistical and transaction databases. The peculiarity rules are defined as a new type of association rules. We described a variant of ER model and semantic networks as a way to represent peculiar data and their relationship among multiple relations (databases). We can change the granularity of the peculiar data dynamically in the discovery process. Some of databases such as Japan-survey, web-log, weather, supermarket have been tested or have been testing for our approach.

Since this project is very new, we just finished the first step. Our future work includes developing a systematic method to mine the rules from multiply databases where there are no explicitly foreign key (link) relationships, and to induce the advanced rules from the RVER models discovered from multiple databases.

References

1. Agrawal R. et al. "Database Mining: A Performance Perspective", *IEEE Trans. Knowl. Data Eng.*, 5(6) (1993) 914-925.
2. Agrawal R. et al. "Fast Discovery of Association Rules", *Advances in Knowledge Discovery and Data Mining*, AAAI Press (1996) 307-328.
3. Aronis, J.M. et al "The WoRLD; Knowledge Discovery from Multiple Distributed Databases", *Proc. 10th International Florida AI Research Symposium (FLAIRS-97)* (1997) 337-341.
4. Fayyad, U.M., Piatetsky-Shapiro, G et al (eds.) *Advances in Knowledge Discovery and Data Mining.* AAAI Press (1996).
5. Freitas, A.A. "On Objective Measures of Rule Surprisingness" J. Zytkow and M. Quafafou (eds.) *Principles of Data Mining and Knowledge Discovery.* Lecture Notes AI 1510, Springer-Verlag (1998) 1-9.
6. Chiang, Roger H.L. et al (eds.) "A Framework for the Design and Evaluation of Reverse Engineering Methods for Relational Databases", *Data & Knowledge Engineering*, Vol.21 (1997) 57-77.
7. Lin, T.Y. "Granular Computing on Binary Relations 1: Data Mining and Neighborhood Systems ", L. Polkowski and A. Skowron (eds.) *Rough Sets in Knowledge Discovery 1*, In Studies in Fuzziness and Soft Computing series, Vol. 18, Physica-Verlag (1998) 107-121.
8. Lin, T.Y., Zhong, N., Dong, J., and Ohsuga, S. "Frameworks for Mining Binary Relations in Data", L. Polkowski and A. Skowron (eds.) *Rough Sets and Current Trends in Computing*, LNAI 1424, Springer-Verlag (1998) 387-393.
9. Liu, B., Hsu W., and Chen, S. "Using General Impressions to Analyze Discovered Classification Rules", *Proc. Third International Conference on Knowledge Discovery and Data Mining (KDD-97)*, AAAI Press (1997) 31-36.
10. Liu, H., Lu H., and Yao, J. "Identifying Relevant Databases for Multidatabase Mining", X. Wu et al. (eds.) *Research and Development in Knowledge Discovery and Data Mining*, Lecture Notes in AI 1394, Springer-Verlag (1998) 210-221.
11. Ribeiro, J.S., Kaufman, K.A., and Kerschberg, L. "Knowledge Discovery from Multiple Databases", *Proc First Inter. Conf. on Knowledge Discovery and Data Mining (KDD-95)*, AAAI Press (1995) 240-245.
12. Silberschatz, A. and Tuzhilin, A. "What Makes Patterns Interesting in Knowledge Discovery Systems", *IEEE Trans. Knowl. Data Eng.*, 8(6) (1996) 970-974.
13. Suzuki E.. "Autonomous Discovery of Reliable Exception Rules", *Proc Third Inter. Conf. on Knowledge Discovery and Data Mining (KDD-97)*, AAAI Press (1997) 259-262.
14. Wrobel, S. "An Algorithm for Multi-relational Discovery of Subgroups", J. Komorowski and J. Zytkow (eds.) *Principles of Data Mining and Knowledge Discovery.* LNAI 1263, Springer-Verlag (1997) 367-375.
15. Yao, Y.Y. "Granular Computing using Neighborhood Systems", Roy, R., Furuhashi, T., and Chawdhry, P.K. (eds.) *Advances in Soft Computing: Engineering Design and Manufacturing*, Springer-Verlag (1999) 539-553.
16. Yao, Y.Y. and Zhong, N. "An Analysis of Quantitative Measures Associated with Rules", Zhong, N. and Zhou, L. (eds.) *Methodologies for Knowledge Discovery and Data Mining*, LNAI 1574, Springer-Verlag (1999) 479-488.
17. Yao, Y.Y. and Zhong, N. "Potential Applications of Granular Computing in Knowledge Discovery and Data Mining", *Proc. The 5th.International Conference on Information Systems Analysis and Synthesis (IASA'99)*, edited in the invited session on Intelligent Data Mining and Knowledge Discovery (1999) (in press).
18. Zadeh, L. A. "Toward a Theory of Fuzzy Information Granulation and Its Centrality in Human Reasoning and Fuzzy Logic", *Fuzzy Sets and Systems*, Elsevier Science Publishers, 90 (1997) 111-127.
19. Zhong N. and Yamashita S. "A Way of Multi-Database Mining", *Proc. the IASTED International Conference on Artificial Intelligence and Soft Computing (ASC'98)*, IASTED/ACTA Press (1998) 384-387.