Available online at www.sciencedirect.com

ScienceDirect

Knowledge-Based Systems 20 (2007) 485–494

Knowledge-Based SYSTEMS

www.elsevier.com/locate/knosys

A rough sets based characteristic relation approach for dynamic attribute generalization in data mining ☆

Tianrui Li [a,b,*], Da Ruan [b,c], Wets Geert [d], Jing Song [a], Yang Xu [a]

[a] *Department of Mathematics, Southwest Jiaotong University, Chengdu, 610031, PR China*
[b] *Belgian Nuclear Research Centre (SCK•CEN), Boeretang 200, 2400 Mol, Belgium*
[c] *Department of Applied Mathematics & Computer Science, Ghent University, Krijgslaan 281 (S9), 9000 Gent, Belgium*
[d] *Department of Applied Economic Sciences, Universiteit Hasselt, 3590 Diepenbeek, Belgium*

Received 21 September 2006; received in revised form 28 December 2006; accepted 18 January 2007
Available online 30 January 2007

Abstract

Any attribute set in an information system may be evolving in time when new information arrives. Approximations of a concept by rough set theory need updating for data mining or other related tasks. For incremental updating approximations of a concept, methods using the tolerance relation and similarity relation have been previously studied in literature. The characteristic relation-based rough sets approach provides more informative results than the tolerance-and-similarity relation based approach. In this paper, an attribute generalization and its relation to feature selection and feature extraction are firstly discussed. Then, a new approach for incrementally updating approximations of a concept is presented under the characteristic relation-based rough sets. Finally, the approach of direct computation of rough set approximations and the proposed approach of dynamic maintenance of rough set approximations are employed for performance comparison. An extensive experimental evaluation on a large soybean database from MLC shows that the proposed approach effectively handles a dynamic attribute generalization in data mining.

Keywords: Rough sets; Knowledge discovery; Data mining; Incomplete information systems

1. Introduction

Data mining, the efficient discovery of previously unknown patterns in large databases, has become a hot topic for decision makers. Rough set theory was originated by Pawlak in 1982 as a formal mathematical theory, modeling knowledge about the domain of interest in terms of a collection of equivalence relations [20]. The main advantage of rough sets is that it does not need any preliminary or additional information about data like probability in probability theory, grade of membership in fuzzy set theory. Nowadays many rough sets based-approaches have been successfully applied in data mining [1,2,5,6,10,14,16,18,19,21,23,24,26–29]. For example, the rough set approach was used to classify different types of meteorological storm events responsible for summer severe weather in [23]. A rough approximation-based clustering to cluster web transactions from web access logs was presented in [5]. Using this approach, users can effectively mine web log records to discover web page access patterns.

Incrementally update knowledge in data mining is getting more and more popular. The volume of data is growing at an unprecedented rate, both in the

☆ This work is partly supported by the National Natural Science Foundation of China with Grant No. 60474022, the Research Fund for the Doctoral Program of Higher Education with Grant No. 20060613007, and China-Flanders bilateral scientific cooperation joint project proposal with Grant Nos. 011S1105 and BOF05B04.

* Corresponding author. Address: Department of Mathematics, Southwest Jiaotong University, Chengdu, 610031, PR China.
E-mail addresses: trli@swjtu.edu.cn, trli30@gmail.com (T. Li), druan@sckcen.be (D. Ruan), geert.wets@uhasselt.be (W. Geert), jesen811206@126.com (J. Song), xuyang@home.swjtu.edu.cn (Y. Xu).

0950-7051/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.knosys.2007.01.002

number of attributes (features) and objects (instances) [13,12,15]. Many databases with genetic information may contain thousands of features for large number of patients. In the technology applications, quantitative (e.g., from sensors) and qualitative (e.g., from manufacturing environment) data from diverse sources may be linked, thus significantly increasing the number of attributes [13]. Under the conventional rough set theory, incrementally mining algorithms for learning classification rules efficiently are discussed in [1,16] when an attribute set in the information system evolves over time. A rough-set-and-rule-tree-based incremental knowledge acquisition algorithm has been proposed in [31] when new objects are added or removed from a given dataset.

The conventional rough set theory is under the assumption that information systems are complete. However, missing data in information systems is common in many real applications. For instance, in the survey sampling, missing data is appearing frequently. It may arise out of poorly designed questionnaires (e.g., inapplicable or ambiguous questions), non-response by an interviewer (e.g., the interviewer does not know or refuses to answer), or errors made by the interviewer (e.g., omitted questions) [7]. Moreover, problems of missing data are especially prevalent in large datasets assembled from several sources. A representative example is the liver transplant database from the National Institute of Diabetes and Digestive and Kidney Diseases [22]. Some fields correspond to real variables with missing data fractions ranging from zero to 100%.

The conventional rough set theory under the indiscernibility relation is limited for analyzing the incomplete information system (IIS). An early extension of rough sets that can directly deal with incomplete data presented by Kryszkiewicz in [12] is under a tolerance relation. The key concept introduced in this method is to associate to the unavailable values of the information system a value of ‘‘null’’ to be considered as a value of ‘‘everything is possible.’’ In [15], a method for incremental updating approximations of a concept in the IIS is proposed under the tolerance relation aiming to a dynamical attribute set. In addition, an algorithm for learning classification rules based on this method has been studied in [17].

Furthermore, Stefanowski et al. extended the model of rough sets by using of a non-symmetric similarity relation in [27,28]. In this approach, objects are described ‘‘incompletely’’ due to our imperfect knowledge and definitely impossible descriptions of all their attributes. Hence, one object x can be considered similar to another object y only if they have the same known values. The lower and upper approximation of a concept obtained using the non-symmetric similarity relation is a refinement of the ones obtained using the tolerance relation [27,28]. An approach for incremental updating approximations of a concept in the IIS is proposed under a non-symmetric similarity relation when a dataset evolves in time by the function of addition and deletion of attributes.

Grzymala-Busse has recently proposed a new extension of rough sets in terms of characteristic relations under the assumption that some of the missing attribute values are lost (e.g., were erased) and some are ‘‘do not care’’ conditions (i.e., they are redundant or unnecessary to make a decision or to classify a case) [7]. This method better reflects the real conditions of IIS than the previous methods. A rule induction algorithm, accepting input data with both lost values and ‘‘do not care’’ conditions, is also described in [8]. In this paper, we discuss how to update approximations of a concept in the IIS through characteristic relations when an attribute set varies with time. Our experimental results have validated the efficiency of the proposed approach. It is crucial to future development of intelligent decision-making systems by rough sets.

The material of the paper is organized as follows. Section 2 introduces basic concepts of rough sets and their extensions as well as some notations used throughout the paper. The concept of attribute generalization and its relation to feature selection and feature extraction are discussed in Section 3. The approach for updating approximations of a concept in the IIS under characteristic relations is illustrated in Section 4. Experimental evaluation of the proposed method is given in Section 5. Section 6 concludes the research work of this paper.

2. Preliminaries

The basic concepts, notations and results of rough sets as well as their extensions are briefly reviewed.

Definition 2.1 [20]. An information system is defined as a pair $\langle U, A\rangle$ where U is a non-empty finite set of objects, $A = C \cup D$ is a non-empty finite set of attributes, C denotes the set of condition attributes and D denotes the set of decision attributes, $C \cap D = \emptyset$. Each attribute $a \in A$ is associated with a set V_a of its value, called the domain of a.

Definition 2.2 [20]. Let $B \subseteq A$ be a subset of attributes. The indiscernibility relation, denoted by I_B, is an equivalence relation defined as

$$\forall x, y, I_B(x, y) \Longleftrightarrow \forall a \in B : a(x) = a(y), \tag{2.1}$$

where $a(x)$ denotes the value of attribute a of objects x.

The classical rough set analysis depends on the indiscernibility relation that describes indistinguishability of objects. Indiscernibility relations are equivalences that are interpreted so that two objects are equivalent if one cannot distinguish them by using existing information.

It is common that missing attribute values exist in real-world information systems due to a variety of causes, e.g., one factor that may contribute to missing data in clinical databases is the expense or difficulty of obtaining certain results, particularly when they are not routine clinical measurements [12]. The classical rough set approach, based on complete information systems, cannot be directly applied in information systems with missing attribute values. An extension of rough sets that can deal with incomplete data presented by Kryszkiewicz in [12].

Definition 2.3 [12]. An information system $\langle U, A\rangle$ is called as an incomplete information system (IIS) if there exists a in A and x in U that satisfy that the value $a(x)$ is unknown, denoted as $*$, which is considered as an "everything is possible" value.

Under this definition of IIS, the toleration and similarity relation are proposed respectively to deal with unknown data in [12,27].

Definition 2.4 [12]. Let $B \subseteq A$ be a subset of attributes. The tolerance relation, denoted by T_B, is defined as:

$$\forall x, y, T_B(x, y) \Longleftrightarrow \forall a \in B : (a(x) = a(y)) \vee (a(y) = *) \vee (a(x) = *). \tag{2.2}$$

Definition 2.5 [27]. Let $B \subseteq A$ be a subset of attributes. The similarity relation, denoted by S_B, is defined as:

$$\forall x, y, S_B(x, y) \Longleftrightarrow \forall a \in B : (a(x) \neq *) \wedge (a(x) = a(y)). \tag{2.3}$$

Obviously, the indiscernibility and tolerance relation are reflexive, transitive and symmetric while similarity relation S is reflexive and transitive, but not symmetric.

However, in many cases, the information system is incomplete owing to the following two reasons [7]. One is that the value of an attribute is lost for a specific case. For instance, it is currently unavailable although it was known originally due to a variety of reasons, e.g., it was recorded but later it was erased. The other reason is the value of an attribute is irrelevant or unimportant. As an example, it is feasible to diagnose a patient in spite of the fact that some clinical test results are not taken. Such missing attribute values do not matter for the final outcome and are called as "do not care" conditions [7]. Under this assumption that missing values in IIS are lost or "do not care" condition, a new version of the IIS is given as follows:

Definition 2.6 [7]. $\langle U, A\rangle$ is an IIS if there exists a in A and x in U that satisfy that the value $a(x)$ is missing. All the missing values are denoted by "?" or "*", where the lost value is denoted by "?", "do not care" condition is denoted by "*".

In [7], the characteristic set and characteristic relation can be determined by using the idea of blocks of attribute-values pairs, which is defined as follows.

Definition 2.7 [7]. Let b be an attribute and v be a value of b for some cases. If t=(b, v) is an attribute-value pair, $v \neq ?$ and $*$, then a block of t, denoted $[t]$, is a set of all cases from U that attribute b have value v. If there exists a case x such that $v = b(x) = ?$, then the case x is not included in the block $[(b, v)]$ for any value v of attribute b. If there exists a case x such that $v = b(x) = *$, then the case x is included in the block $[(b, v)]$ for all value v of attribute b.

Example 2.8. Table 1 contains the following incomplete information table, which will be further used in the paper to illustrate the above concepts, where the attribute set A=$\{a, b, c, d\}$, the object set U=$\{1, 2, \ldots, 8\}$ and "?" and "*" represent lost values and "do not care" conditions, respectively.

Then from Definition 6, we have the following results:

[(a,0)]={2,3,4,6}, [(a,1)]={4,7,8},
[(b,0)]={1,4,5}, [(b,1)]={2,3,6},
[(c,1)]={1,4}, [(c,2)]={2,3,4,6}, [(c,0)]={4,7},
[(d,0)]={1,2,3,7}, [(d,1)]={4,5,6,7},
[(d,2)]={7,8}.

Definition 2.9. [7] Let $B \subseteq A$ be a subset of attributes. The characteristic set $I_B^C(x)$ is the intersection of blocks of attribute-value pairs (b, v) for all attributes b from B for which $b(x)$ is specified and $b(x) = v$.

Obviously, characteristic sets are the generalization of elementary sets in complete information systems.

Example 2.10. For Table 1 and $B = A$, we have

$I_B^C(1) = \{1,4,5\} \cap \{1,4\} \cap \{1,2,3,7\} = \{1\}$,
$I_B^C(2) = \{2,3,4,6\} \cap \{2,3,6\} \cap \{2,3,4,6\} \cap \{1,2,3,7\} = \{2,3\}$,
$I_B^C(3) = \{2,3,4,6\} \cap \{2,3,6\} \cap \{2,3,4,6\} \cap \{1,2,3,7\} = \{2, 3\}$,
$I_B^C(4) = U \cap \{1,4,5\} \cap \{1,2,3,4,6\} \cap \{4,5,6,7\} = \{4\}$,
$I_B^C(5) = \{1,4,5\} \cap \{4,5,6,7\} = \{4,5\}$,
$I_B^C(6) = \{2,3,4,6\} \cap \{2,3,6\} \cap \{2,3,4,6\} \cap \{4,5,6,7\} = \{6\}$,
$I_B^C(7) = \{4,7,8\} \cap \{4,7\} \cap U = \{4,7\}$,
$I_B^C(8) = \{4,7,8\} \cap \{7,8\} = \{7,8\}$.

By the definition of characteristic set, the characteristic relation in the IIS is given below [7].

Definition 2.11. Let $B \subseteq A$ be a subset of attributes. The characteristic relation, denoted by C_B, is defined as: $(x, y) \in C_B \Longleftrightarrow y \in I_B^C(x)$.

Table 1
An incomplete information system

U	a	b	c	d
1	?	0	1	0
2	0	1	2	0
3	0	1	2	0
4	*	0	*	1
5	?	0	?	1
6	0	1	2	1
7	1	?	0	*
8	1	?	?	2

The characteristic relation C_B is reflexive but not symmetric and transitive. Obviously, it is a generalization of the indiscernibility relation in complete information systems.

Definition 2.12. The lower and upper approximations of X with regard to B under the characteristic relation are

$$X_B^C = \cup\{I_B^C(x)|x \in X, I_B^C(x) \subseteq X\}, \tag{2.4}$$

$$X_C^B = \cup\{I_B^C(x)|x \in X, I_B^C(x) \cap X \neq \emptyset\} = \cup\{I_B^C(x)|x \in X\}, \tag{2.5}$$

respectively.

Remark 2.13. We only use this definition of approximations, namely, *concept* approximations in [7], since propositions in Section 4 do not hold under the definitions of *singleton* and *subset* approximations.

Definition 2.14. The lower and upper boundary sets of X with regard to B under characteristic relation are defined as $\Delta X_B^C = X - X_B^C$ and $\Delta X_C^B = X_C^B - X$, respectively.

Example 2.15. Let $X = \{1,2,4,7,8\}$, $B = A$. Then from Table 1, we have $X_B^C = \{1,4,7,8\}$, $X_C^B = \{1,2,3,4,5,7,8\}$, $\Delta X_B^C = \{2\}$, $\Delta X_C^B = \{3,5\}$.

3. Attribute generalization

In this section, a concept of an attribute generalization and related issues of feature selection and feature extraction are first given. Then the relation among them is discussed. An example from clinical decision making is to illustrate the attribute generalization in real-life applications.

Definition 3.1. Attribute generalization refers to dynamic changing of the attribute set in an information system according to the need of real-life applications.

In the age of an information explosion, attribute generalization is an important research topic of data mining and knowledge discovery in database. In many domains, e.g., clinical decision making, intrusion detection, stock evaluation, and text categorization, one collects many features that are potentially useful. However, all of these features may not be useful or relevant to one's classification, forecasting, or clustering objects. Therefore, deleting unnecessary features in the original feature set will often lead to a better performance. For instance, although there may be dozens of features (make, brand, year, weight, length, height, engine size, transmission, color, owner, and price.) available when one purchases a second hand vehicle, one may only read a handful of important features (e.g., make, year, engine, color and price) that meet one's needs [31]. In addition, the collection of features is not static due to the need of real-life applications, e.g., construction of features in an intrusion detection system. Therefore, adding necessary features in the original feature set will generally enhance the accuracy of classification as well as its effectiveness. For example, time-window based features, e.g., number of connections from the source IP to the same destination port in the last T seconds, in the intrusion detection system are constructed and added to the original feature set to capture new scanning attacks [4].

Feature selection and feature extraction, widely discussed in statistics and pattern recognition literature, are two kinds of approaches of dimensionality reduction for classifications [3,11].

Definition 3.2. [3] Feature selection is a process to find the optimal subset of features that satisfies certain criteria and only removes the features that are unnecessary or unimportant to the target concept and the remaining features are kept intact.

Feature selection addresses the specific task of finding a subset of given features that are useful to solve the domain problem, without disrupting the underlying meaning of the selected features. It reduces the dimensionality of the data and may allow learning algorithms to operate faster and more effectively. In some cases, accuracy on future classification can be improved; in others, the result is a more compact and easily interpreted representation of the target concept [5,9,30].

Definition 3.3. [11] Feature extraction creates new features by irreversibly transforming the original features such that the created features contain most useful information for the target concept.

The process of feature extraction is more complicated. It is difficult to compare the effectiveness of feature selection and feature extraction as they are employed under different circumstances [11].

Obviously feature selection and feature extraction may be considered as two steps in an attribute generalization, in other words, the attribute generalization involves feature selection and feature extraction. Among many applications in different domains, clinical decision making, is a successful example of the attributed generalization.

Example 3.4. In recent years, the use of digital technology has supported widespread sharing of electronic health care data. Those data are generally stored in clinical information systems. Clinical decision making has focused on improving clinicians' diagnostic accuracy by using a variety of techniques like rough set theory, decision tree through these information systems. Table 2 is an information system about the symptoms of patients collected for identifying the heart disease in a hospital.

where

Sex: {1: male, 0: female}
CP (Chest pain): {3: normal angina, 2: atypical angina, 1: non-anginal pain, 0: asymptomatic}
S (Snuffle): {1: yes, 0: no}

Table 2
Incomplete information system of heart disease

Patients	Sex	CP	F	S	C	BP	HR	D	HD
1	0	?	1	0	0	0	?	1	1
2	1	1	0	1	1	2	2	1	0
3	0	3	0	1	1	0	2	1	0
4	1	*	1	0	*	1	*	0	1
5	0	?	1	*	?	1	2	*	1
6	1	2	0	1	1	2	?	1	1
7	1	1	?	1	0	*	1	0	0
8	0	0	?	0	?	1	0	1	0

F (Fatigue): {1: yes, 0: no}
C (Cough): {1: yes, 0: no}
BP (Blood pressure): {2: high, 1: normal, 0: low}
HR (Heart rate): {2: high, 1: normal, 0: low}
D (Dyspnea, means difficulty in breathing): {1: yes, 0: no}
HD (Heart disease): {1: yes, 0: no}

In Table 2, symptom “snuffle” may be recorded in this information system at the beginning. However, according to the expert opinions or other decision making approaches, symptom “snuffle” is independent of heart disease. Therefore, snuffle is an outdated or useless information for this case and can be deleted from the information system (see Table 3) to reduce computational complexity of knowledge discovery.

In addition, the information system generally is not static in the medical practice when new examination tools are used or more tests are ordered by physicians for enhancement of accuracy of medical diagnosis. Therefore, new clinical information, e.g., symptoms “syncope, electrocardiographic,” may come (see Table 4).

where

Sex: {1: male, 0: female}
S (Syncope, means a brief loss of consciousness): {1: yes, 0: no}
EC (Electrocardiographic): {2: normal, 1: ST-T wave abnormality, 0: hypertrophy}
Heart disease: {1: yes, 0: no}

These information systems (namely, Tables 3 and 4) will be used for future better clinical decision making. Table 5 is an incorporated information system of Tables 3 and 4.

Table 3
Revised incomplete information system of heart disease

Patients	Sex	CP	F	C	BP	HR	D	HD
1	0	?	1	0	0	?	1	1
2	1	1	0	1	2	2	1	0
3	0	3	0	1	0	2	1	0
4	1	*	1	*	1	*	0	1
5	0	?	1	?	1	2	*	1
6	1	2	0	1	2	?	1	1
7	1	1	?	0	*	1	0	0
8	0	0	?	?	1	0	1	0

Table 4
New clinical information of heart disease

Patients	Sex	S	EC	Heart disease
1	0	?	1	1
2	1	0	2	0
3	0	1	2	0
4	1	*	*	1
5	0	?	?	1
6	1	1	2	1
7	1	0	?	0
8	0	1	0	0

Table 5
New incomplete information system of heart disease after the attribute generalization

Patients	Sex	CP	F	C	BP	HR	EC	S	D	HD
1	0	?	1	0	0	?	1	?	1	1
2	1	1	0	1	2	2	2	0	1	0
3	0	3	0	1	0	2	2	1	1	0
4	1	*	1	*	1	*	*	*	0	1
5	0	?	1	?	1	2	?	?	*	1
6	1	2	0	1	2	?	2	1	1	1
7	1	1	?	0	*	1	?	0	0	0
8	0	0	?	?	1	0	0	1	1	0

Remark 3.5. In this paper, we do not pay much attention to the methods on attribute generalization while emphasize how to use rough set methodology to incrementally update approximations of a concept under characteristic relations when the attribute generalization happens in the IIS.

4. Dynamic maintenance of rough set approximations under characteristic relations

Let X be a subset of U. The following propositions show that we can incrementally update approximations of a concept in the IIS based on characteristic relations, namely, the lower and upper approximations of X can be updated by using the original information.

Lemma 4.1. *Let $P, Q \subseteq A$. Then $X^C_P \subseteq X^C_{P\cup Q}$, $X^C_Q \subseteq X^C_{P\cup Q}$.*

Example 4.2. Let $X = \{1,2,4,7,8\}$, $P = \{a,c\}$, $Q = \{b,d\}$. Then from Table 1, we have $X^C_{P\cup Q} = \{1,4,7,8\}$, $X^C_P = \{1,7,8\}$. Obviously, $X^C_P \subseteq X^C_{P\cup Q}$.

Lemma 4.3. *Let $P, Q \subseteq A$. Then $\Delta X^C_{P\cup Q} \subseteq \Delta X^C_P$, $\Delta X^{P\cup Q}_C \subseteq \Delta X^P_C$.*

Example 4.4. Let $X = \{1,2,4,7,8\}$, $P = \{a,c\}$, $Q = \{b,d\}$. Then from Table 1, we have the following results:

$$X^C_{P\cup Q} = \{1,4,7,8\},\ X^{P\cup Q}_C = \{1,2,3,4,7,8\},$$
$$X^C_P = \{1,7,8\},\ X^P_C = \{1,2,3,4,6,7,8\},$$
$$\Delta X^C_{P\cup Q} = \{2\} \subseteq \Delta X^C_P = \{2,4\},$$
$$\Delta X^{P\cup Q}_C = \{3\} \subseteq \Delta X^P_C = \{3,6\}.$$

Proposition 4.5. *Let* $P, Q \subseteq A$ *and* $Q \cap P = \emptyset$. *Then* $X^C_{P\cup Q} = X^C_P \cup X^C_Q \cup Y$, *where* $Y = \{x \in \Delta X^C_P \cap \Delta X^C_Q | \cap_{a\in P\cup Q}. I^C_{\{a\}}(x) \subseteq X\}$.

Proof. Let $X \subset U$, $x \in U$ such that $x \in X^C_{P\cup Q}$. From Lemma 4.1, we have $X^C_P \subseteq X^C_{P\cup Q}$, $X^C_Q \subseteq X^C_{P\cup Q}$, $x \in X^C_{P\cup Q} \Longleftrightarrow \cap_{a\in P\cup Q} I^C_{\{a\}}(x) \subseteq X$. Therefore if $x \notin X^C_P \cup X^C_Q$, we have $x \in \Delta X^C_P \cap \Delta X^C_Q$, namely $x \in Y$. So we can get $X^C_{P\cup Q} \subseteq X^C_P \cup X^C_Q \cup Y$. In addition, $X^C_{P\cup Q} \supseteq X^C_P \cup X^C_Q \cup Y$. Therefore, we have $X^C_{P\cup Q} = X^C_P \cup X^C_Q \cup Y$. □

Example 4.6. Let $X = \{1,2,4,7,8\}$, $P = \{a,b\}$, $Q = \{c,d\}$, $R = P \cup Q$. Then from Table 1, we have $X^C_P = \{4,7,8\}$, $\Delta X^C_P = \{1,2\}$, $X^C_Q = \{1,4,7,8\}$, $\Delta X^C_Q = \{2,4\}$ and $\Delta X^C_P \cap \Delta X^C_Q = \{2\}$. Since $\cap_{a\in P\cup Q} I^C_a(2) = \{2,3,4,6\} \cap \{2,3,6\} \cap \{2,3,4,6\} \cap \{1,2,3,7\} = \{2,3\} \not\subset X$, we have $Y = \{x \in \Delta X^C_P \cap \Delta X^C_Q | \cap_{a\in P\cup Q}. I^C_{\{a\}}(x) \subseteq X\} = \emptyset$. Hence, we get $X^C_R = X^C_P \cup X^C_Q \cup Y = \{4,7,8\} \cup \{1,4,7,8\} \cup \emptyset = \{1,4,7,8\}$.

Proposition 4.7. *Let* $Q \subset P \subseteq A$. *Then* $X^C_{P-Q} = X^C_P - \Delta X^C_{P-Q}$, *where* $\Delta X^C_{P-Q} = \{x \in \cap_{a\in P-Q} \Delta X^C_{\{a\}} | \cap_{a\in P-Q}. I^C_{\{a\}}(x) \not\subset X\}$.

Proof. Obviously $X^C_{P-Q} \subseteq X^C_P, \Delta X^C_P \subseteq \Delta X^C_{P-Q}, X^C_{P-Q} \cup \Delta X^C_{P-Q} = X = X^C_P \cup \Delta X^C_P$. Thus we have $X^C_{P-Q} = X^C_P \cup \Delta X^C_P - \Delta X^C_{P-Q} = X^C_P - \Delta X^C_{P-Q}$. □

Example 4.8. Let $X = \{1,2,4,7,8\}$, $P = \{a,b,c,d\}$, $Q = \{b,c\}$ and $R = P - Q = \{a,d\}$. Then from Table 1, we have $X^C_P = \{1,4,7,8\}$. Since $X^C_{\{a\}} = \{1,4,7,8\}$, $\Delta X^C_{\{a\}} = \{2\}$, $X^C_{\{d\}} = \{7,8\}$, $\Delta X^C_{\{d\}} = \{1,2,4\}$, then $\Delta X^C_{\{a\}} \cap \Delta X^C_{\{d\}} = \{2\}$. In addition, $I^C_{\{a\}}(2) \cap I^C_{\{d\}}(2) = \{2, 3\} \not\subset X$. Thus, $\Delta X^C_{P-Q} = \{x \in \cap_{a\in P-Q} \Delta X^C_{\{a\}} | \cap_{a\in P-Q}. I^C_{\{a\}}(x) \not\subset X\} = \{1,2\}$. Therefore, we have $X^C_{P-Q} = X^C_P - \Delta X^C_{P-Q} = \{4,7,8\}$.

Proposition 4.9. *Let* $P, Q \subseteq A$ *and* $Q \cap P = \emptyset$. *Then* $X^{P\cup Q}_C = X^P_C \cap X^Q_C - Z$, *where* $Z = \{x \in \cap_{a\in P\cup Q} \Delta X^{\{a\}}_C | \cap_{a\in P\cup Q}. I^C_{\{a\}}(x) \subseteq \cap_{a\in P\cup Q} \Delta X^{\{a\}}_C\}$.

Proof. Let $x \in X^P_C \cap X^Q_C - Z$ and $x \notin X$. If $\cap_{a\in P\cup Q} I^C_{\{a\}}(x) \cap X = \emptyset$, then we have $\cap_{a\in P\cup Q} I^C_{\{a\}}(x) \subseteq \Delta X^{\{a\}}_C$, $\forall a \in P - Q$. Thus $\cap_{a\in P\cup Q} I^C_{\{a\}}(x) \subseteq \cap_{a\in P\cup Q} \Delta X^{\{a\}}_C$, namely, $x \in Z$, which contradicts the assumption that $x \in X^P_C \cap X^Q_C - Z$. Therefore $X^{P\cup Q}_C \supseteq X^P_C \cap X^Q_C - Z$. □

On the other hand, $\Delta X^P_C \supseteq \Delta X^{P\cup Q}_C$, $\Delta X^Q_C \supseteq \Delta X^{P\cup Q}_C$. Let $x \in X^{P\cup Q}_C$ and $x \notin X$. Then we have $x \in \Delta X^P_C \cap \Delta X^Q_C$, $\cap_{a\in P\cup Q} I^C_{\{a\}}(x) \cap X \neq \emptyset$. Because $X \cap \cap_{a\in P\cup Q} \Delta X^{\{a\}}_C = \emptyset$, this implies that $\cap_{a\in P\cup Q} I^C_{\{a\}}(x)$ is not a subset of $\cap_{a\in P\cup Q} \Delta X^{\{a\}}_C$. Thus $x \notin z$. Therefore, we can get $x \in \Delta X^P_C \cap \Delta X^Q_C - Z$, namely, $X^{P\cup Q}_C \subseteq X \cup (\Delta X^P_C \cap \Delta X^Q_C - Z)$. So we have $X^{P\cup Q}_C = X^P_C \cap X^Q_C - Z$.

Example 6. Let $X = \{1,2,4,7,8\}$, $P = \{a,c\}$, $Q = \{b,d\}$ and $R = P \cup Q$. Then from Table 1, we have $X^P_C = \{1,2,3,4,6,7,8\}$, $X^Q_C = \{1,2,3,4,5,7,8\}$. Since $\Delta X^{\{a\}}_C = \Delta X^{\{c\}}_C = \{3,6\}$, $\Delta X^{\{b\}}_C = \Delta X^{\{d\}}_C = \{3,5,6\}$, thus $\Delta X^{\{a\}}_C \cap \Delta X^{\{b\}}_C \cap \Delta X^{\{c\}}_C \cap \Delta X^{\{d\}}_C = \{3,6\}$. In addition,

$$\cap_{a\in P\cup Q} I^C_{\{a\}}(3) = \{2,3,4,6\} \cap \{2,3,6\} \cap \{2,3,4,6\} \cap \{1,2,3,7\} = \{2,3\} \not\subset \cap_{a\in P\cup Q} \Delta X^{\{a\}}_C.$$

Similarly, we have

$$\cap_{a\in P\cup Q} I^C_{\{a\}}(6) = \{2,3,4,6\} \cap \{2,3,6\} \cap \{2,3,4,6\} \cap \{4,5,6,7\} = \{6\} \subseteq \cap_{a\in P\cup Q} \Delta X^{\{a\}}_C.$$

Then $Z = \{6\}$. Therefore, we have $X^{P\cup Q}_C = X^P_C \cap X^Q_C - Z = \{1,2,3,4,7,8\} - \{6\} = \{1,2,3,4,7,8\}$.

Proposition 4.10. *Let* $Q \subset P \subseteq A$. *Then* $X^{P-Q}_C = X^P_C \cup Z'$, *where* $Z' = \{x \in \cap_{a\in P-Q} \Delta X^{\{a\}}_C | \cap_{a\in P-Q}. I^C_{\{a\}}(x) \not\subset \cap_{a\in P-Q} \Delta X^{\{a\}}_C\}$.

Proof. Let $x \in Z'$. If $\cap_{a\in P-Q} I^C_{\{a\}}(x) \cap X = \emptyset$, then $\cap_{a\in P-Q} I^C_{\{a\}}(x) \subseteq \Delta X^{\{a\}}_C, \forall a \in P - Q$. Thus $\cap_{a\in P-Q} I^C_{\{a\}}(x) \subseteq \cap_{a\in P-Q} \Delta X^{\{a\}}_C$, namely, $x \notin Z'$, which contradicts the assumption $x \in Z'$. So we have $\cap_{a\in P-Q} I^C_{\{a\}}(x) \cap X \neq \emptyset$. Then $x \in X^{P-Q}$. Thus we get $X^{P-Q}_C \supseteq X^P_C \cup Z'$. □

On the other hand, $X^{P-Q}_C = X \cup \Delta X^{P-Q}_C$, $\Delta X^P_C \subseteq \Delta X^{P-Q}_C$. Let $x \in X^{P-Q}_C$ and $x \notin X^P_C$. Because $\Delta X^{P-Q}_C \subseteq \Delta X^{\{a\}}_C$, $\forall a \in P - Q$, we have $\Delta X^{P-Q}_C \subseteq \cap_{a\in P-Q} \Delta X^{\{a\}}_C$. Thus $x \in \cap_{a\in P-Q} \Delta X^{\{a\}}_C$. If $\cap_{a\in P-Q} I^C_{\{a\}}(x) \subseteq \cap_{a\in P-Q} \Delta X^{\{a\}}_C$, because $\cap_{a\in P-Q} \Delta X^{\{a\}}_C \cap X = \emptyset$, then we have $\cap_{a\in P-Q} I^C_{\{a\}}(x) \cap X = \emptyset$. Therefore, $x \notin X^{P-Q}_C$. This is a contradiction according to the assumption that $x \in X^{P-Q}_C$. So $x \in Z'$. Therefore, we have $X^{P-Q}_C \subseteq X^P_C \cup Z'$ and $X^{P-Q}_C = X^P_C \cup Z'$.

Example 4.11. Let $X = \{1,2,4,7,8\}$, $P = \{a,b,c,d\}$, $Q = \{b,c\}$ and $R = P - Q = \{a,d\}$. From Table 1, we have $X^P_C = \{1,2,3,4,7,8\}$, $\Delta X^P_C = \{3\}$, $\Delta X^{\{c\}}_C \cap \Delta X^{\{d\}}_C = \{3,6\}$. Since $\cap_{a\in P-Q} I^C_{\{a\}}(3) = \{2,3,4,6\} \cap \{1,2,3,7\} = \{2,3\} \not\subset \Delta X^{\{c\}}_C \cap \Delta X^{\{d\}}_C$. Similarly, we have $\cap_{a\in P-Q} I^C_{\{a\}}(6) = \{4,6\} \not\subset \Delta X^{\{c\}}_C \cap \Delta X^{\{d\}}_C$. Thus $Z' = \{3,6\}$. Therefore, we have $X^{P-Q}_C = X^P_C \cup Z' = \{1,2,3,4,5,7,8\} \cup \{3,6\} = \{1,2,3,4,6,7,8\}$.

Remark 4.12. The above propositions show that we can really realize dynamic maintenance of rough set approximations in the IIS based on the characteristic relation by using the boundary sets of single attributes and the intersection of the set denoted by corresponding attribute-value pairs.

5. Experimental evaluation

Experiments were performed on a 400 MHz Pentium Server with 256 MB of memory, running windows 2000 server and SQL server 2000. Algorithms were coded in C#. We chose the large soybean database, publicly available from the UC Irvine Machine Learning Database Repository (www.ics.uci.edu/~mlearn/MLRepository.html), as a benchmark dataset for the performance tests. The reason for selecting this database in our experimental evaluation is that there are two attribute values, "dna" (means does not apply) and "?" (an unknown value), in this database that can be regarded as "do not care" condition and the lost value respectively in the IIS. It is convenient for the experimental evaluation of the proposed approaches. The large soybean database consists of 683 instance (objects). The number of attributes, all having been nominalized, is 35. There are 19 classes, namely, diaporthe-stem-canker, charcoal-rot, rhizoctonia-root-rot, phytophthora-rot, brown-stem-rot, powdery-mildew, downy-mildew, brown-spot, bacterial-blight, bacterial-pustule, purple-seed-stain, anthracnose, phyllosticta-leaf-spot, alternarialeaf-spot, frog-eye-leaf-spot, dia-porthe-pod-&-stem-blight, cyst-nematode, 2-4-d-injury, and herbicide-injury. In our experiments, the 19 classes are denoted as $X_1,\ldots,X_{19}$, respectively.

Table 6
The number of objects in the approximations and boundary sets of 19 classes

Classes	N_1	N_2	N_3	N_4
X_1	28	16	8	4
X_2	20	20	0	0
X_3	20	20	0	0
X_4	93	85	5	3
X_5	45	42	1	2
X_6	22	19	2	1
X_7	21	19	1	1
X_8	96	84	4	8
X_9	21	19	1	1
X_{10}	23	18	3	2
X_{11}	20	20	0	0
X_{12}	44	44	0	0
X_{13}	20	20	0	0
X_{14}	99	82	8	9
X_{15}	100	85	9	6
X_{16}	18	9	3	6
X_{17}	14	14	0	0
X_{18}	683	0	667	16
X_{19}	22	0	14	8

5.1. A comparison of the number of objects in the approximations of classes with that of their corresponding boundary sets

Usually, the target size determines the calculation speed. The greater number will be relatively time-consuming. The characteristics of the proposed approaches to compute the approximations of classes are in virtue of their corresponding boundary sets. Therefore, in the first experiment, we proceed a comparison of the number of objects in the approximations of classes with that of their corresponding boundary sets. The following experimental results show that the number of objects in the boundary sets is generally far less than that in the corresponding approximations of classes.

Firstly, let A denote all 35 attributes in the database. Then, the lower and upper approximations of $X_1,\ldots,X_{19}$ with regard to A under the characteristic relation are themselves. Hence, the number of objects in the boundary sets of 19 classes is 0, which is obviously far less than that in the corresponding approximations.

Secondly, we randomly select 11 attributes from the database, namely, $B =$ "date, plant_stand, precip, temp, hail, crop_hist, severity, seed_tmt, stem_cankers, canker_ lesion, sclerotia, fruit_pods, seed." The number of objects in the approximations and boundary sets of 19 classes with regard to B under the characteristic relation is shown in Table 6.

where

N_1: The number of objects in its upper approximation.
N_2: The number of objects in its lower approximation.
N_3: The number of objects in its upper boundary set.
N_4: The number of objects in its lower boundary set.

From Table 6, we obtain the following results:

(1) In most classes (15 out of 19 classes), there is a sharp difference between the number of their approximations and that of their corresponding boundary sets.
(2) In these 2 classes, X_1, X_{16} (namely, " diaporthe-stem-canker, diaporthe-pod-&-stem-blight"), the number of their boundary sets occupies for nearly half of the number of their corresponding approximations.
(3) Only in the classes X_{18}, X_{19} (namely, "2-4-d-injury, herbicide-injury"), there is a little difference between the number of its approximations and that of its corresponding boundary sets.

Remark 5.1. This fact shows that the complexity of the proposed method will be reduced since the main operation of computing approximations of classes is related to the number of their corresponding boundary sets.

5.2. Performance evaluation of dynamic maintenance of rough set approximations

We employ the approach of direct computation of rough set approximations (denoted as DCRSA) and the proposed approach of dynamic maintenance of rough set approximations (denoted as IDCRSA) for a performance comparison. We continue the second experiment (divided into three separate parts) to validate the effectiveness of IDCRSA. From the following experimental results, clearly IDCRSA outperforms DCRSA in different cases that validates the proposed method is efficient for dynamic maintenance of rough set approximations in the IIS.

5.2.1. A comparison of DCRSA with IDCRSA when attributes are added to the original attribute set

We randomly select two attribute sets P and Q ($P \cap Q = \emptyset$) and one object set (namely, diaporthe-stem-canker) in our experiments.

5.2.1.1. A comparison of DCRSA with IDCRSA when calculating lower approximations. The runtime of calculating lower approximations of class "diaporthe-stem-canker" employing DCRSA and IDCRSA is listed in Table 7 when attributes are added to the original attribute set.

where

$P_1 =$ {date, plant_stand, precip, temp, hail, crop_hist, area_damaged, severity, seed_tmt, germination, plant_growth, leaves, leafspots_halo, leafspots_marg, leafspot_size}.

$Q_1 =$ {leaf_shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers}.

$P_2 =$ {canker_lesion, fruiting_bodies, external_decay, mycelium, int_discolor, sclerotia, fruit_pods}.

$Q_2 =$ {fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$P_3 =$ {date, plant_stand, precip, temp, hail, crop_hist, area_damaged, severity, seed_tmt, germination, plant_growth, leaves, leafspots_halo, leafspots_marg, leafspot_size, leaf_ shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers}.

$Q_3 =$ {canker_lesion, fruiting_bodies, external_decay, mycelium, int_discolor, sclerotia, fruit_pods, fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$P_i \cup Q_i, i = 1, 2, 3$ means that the attribute set Q_i is added into P_i.

5.2.1.2. A comparison of DCRSA with IDCRSA when calculating upper approximations. The runtime of calculating upper approximations of class "diaporthe-stem-canker" employing DCRSA and IDCRSA is listed in Table 8 when attributes are added to the original attribute set.

where $P_i, Q_i, i = 1, 2, 3$ are the same as those in Table 7.

Table 7
A comparison of DCRSA with IDCRSA when calculating lower approximations (second)

	$P_1 \cup Q_1$	$P_2 \cup Q_2$	$P_3 \cup Q_3$
DCRSA	309	1674	255
IDCRSA	<2	3	<2

Table 8
A comparison of DCRSA with IDCRSA when calculating upper approximations (second)

	$P_1 \cup Q_1$	$P_2 \cup Q_2$	$P_3 \cup Q_3$
DCRSA	307	1670	251
IDCRSA	38	97	91

5.2.2. A comparison of DCRSA with IDCRSA when attributes are deleted from the original attribute set

We randomly select two attribute sets P and Q ($Q \subset P$) and one object set (namely, diaporthe-stem-canker) in our experiments.

5.2.2.1. A comparison of DCRSA with IDCRSA when calculating lower approximations. The runtime of calculating lower approximations of class "diaporthe-stem-canker" employing DCRSA and IDCRSA is listed in Table 9 when attributes are deleted from the original attribute set.

where

$P_4 =$ {date, plant_stand, precip, temp, hail, crop_hist, area_damaged, severity, seed_tmt, germination, plant_growth, leaves, leafspots_halo, leafspots_marg, leafspot_size, leaf_ shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers}.

$Q_4 =$ {leaf_shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers}.

$P_5 =$ {canker_lesion, fruiting_bodies, external_decay, mycelium, int_discolor, sclerotia, fruit_pods, fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$Q_5 =$ {fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$P_6 =$ {date, plant_stand, precip, temp, hail, crop_hist, area_damaged, severity, seed_tmt, germination, plant_growth, leaves, leafspots_halo, leafspots_marg, leafspot_size, leaf_ shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers, canker_lesion, fruiting_ bodies, external_decay, mycelium, int_discolor, sclerotia, fruit_pods, fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$Q_6 =$ {canker_lesion, fruiting_bodies, external_decay, mycelium, int_discolor, sclerotia, fruit_pods, fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$P_i \backslash Q_i, i = 4, 5, 6$ means that the attribute set Q_i is deleted from P_i.

Table 9
A comparison of DCRSA with IDCRSA when calculating lower approximations (second)

	$P_4 \backslash Q_4$	$P_5 \backslash Q_5$	$P_6 \backslash Q_6$
DCRSA	579	1434	312
IDCRSA	43	116	79

Table 10
A comparison of DCRSA with IDCRSA when calculating upper approximations (second)

	$P_4 \backslash Q_4$	$P_5 \backslash Q_5$	$P_6 \backslash Q_6$
DCRSA	623	1570	268
IDCRSA	39	102	56

5.2.2.2. A comparison of DCRSA with IDCRSA when calculating upper approximations. The runtime of calculating upper approximations of class “diaporthe-stem-canker” employing DCRSA and IDCRSA is listed in Table 10 when attributes are deleted from the original attribute set.

where $P_i, Q_i, i = 1,2,3$ are the same as those in Table 9.

5.2.3. A comparison of DCRSA with IDCRSA when an attribute generalization happens

We randomly select two attribute sets P and Q ($Q \cap P \neq \emptyset$) and one object set (namely, diaporthe-stem-canker) in our experiments.

5.2.3.1. A comparison of DCRSA with IDCRSA when calculating lower approximations. The runtime of calculating lower approximations of class “diaporthe-stem-canker” employing DCRSA and IDCRSA is listed in Table 11 when an attribute generalization happens.

where

$P_7 =$ {date, plant_stand, precip, temp, hail, crop_hist, area_damaged, severity, seed_tmt, germination, plant_growth, leaves, leafspots_halo, leafspots_marg, leafspot_size, leaf_ shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers}.

$Q_7 =$ {leaf_shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers, fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$P_8 =$ {canker_lesion, fruiting_bodies, external_decay, mycelium, int_discolor, sclerotia, fruit_pods, fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots}.

$Q_8 =$ {fruit_spots, seed, mold_growth, seed_discolor, seed_size, shriveling, roots, leaf_ shread, leaf_malf, leaf_mild, stem, lodging, stem_cankers}.

$P_i \cup Q_i, i = 7,8$ means that the attribute set Q_i is merged with P_i.

5.2.3.2. A comparison of DCRSA with IDCRSA when calculating upper approximations. The runtime of calculating upper approximations of class “diaporthe-stem-canker” employing DCRSA and IDCRSA is listed in Table 12 when an attribute generalization happens.

where $P_i, Q_i, i = 7,8$ are the same as those in Table 11.

Table 11
A comparison of DCRSA with IDCRSA when calculating lower approximations (second)

	$P_7 \cup Q_7$	$P_8 \cup Q_8$
DCRSA	402	577
IDCRSA	44	156

Table 12
A comparison of DCRSA with IDCRSA when calculating upper approximations (second)

	$P_7 \cup Q_7$	$P_8 \cup Q_8$
DCRSA	395	404
IDCRSA	71	175

6. Conclusions

Based on the extension of the classical rough set theory to the IIS under characteristic relations, we realized updating approximation of a concept incrementally by adding and removing some attributes simultaneously in the IIS, which is crucial to mining tasks when knowledge updates. An extensive set of experiments confirms that the proposed approach is effective for dynamic maintenance of rough set approximations. An interesting direction of future work is to study how to use this feature to develop algorithms as well as a system for learning certain and possible classification rules incrementally from the IIS through the characteristic relation. It also seems worthwhile to explore if the proposed approach can be extended to other generalized rough set models such as fuzzy rough set theory [25].

References

[1] C.C. Chan, A rough set approach to attribute generalization in data mining, Information Sciences 107 (1998) 177–194.

[2] L. Chang, G. Wang, Y. Wu, An approach for attribute reduction and rule generation based on rough set theory, Journal of Software 10 (11) (1999) 1206–1211.

[3] M. Dash, H. Liu, Feature selection for classification, Intelligence Data Analysis 1 (1997) 131–156.

[4] O. Depren, M. Topallar, E. Anarim, M.K. Ciliz, An intelligent intrusion detection system (ids) for anomaly and misuse detection in computer networks, Expert Systems with Applications 29 (4) (2005) 713–722.

[5] J.G. Dy, C.E. Brodley, Feature selection for unsupervised learning, The Journal of Machine Learning Research archive 5 (2004) 845–889.

[6] S.J. Fakih, T.K. Das, LEAD: A methodology for learning efficient approaches to medical diagnosis, IEEE Transactions on Information Technology in Biomedicine 10 (2) (2006) 220–228.

[7] J.W. Grzymala-Busse, Characteristic relations for incomplete data: A generalization of the indiscernibility relation, Transactions on Rough Sets IV (2005) 58–68.

[8] J.W. Grzymala-Busse, S. Siddhaye, Rough set approaches to rule induction from incomplete data, in: International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems, 2004, pp. 923–930.

[9] M.A. Hall, Correlation-based feature selection for machine learning, Ph.D. thesis, Waikato University, New Zealand, 1999.

[10] T.P. Hong, L.H. Tseng, S.L. Wang, Learning rules from incomplete training examples by rough sets, Expert System with Applications 22 (4) (2002) 285–293.

[11] J. Kittler, Feature selection and extraction, in: T.Y. Young, K. Fu (Eds.), Handbook of Pattern Recognition and Image Processing, Academic Press, New York, 1986, pp. 203–217.

[12] M. Kryszkiewicz, Rough set approach to incomplete information system, Information Sciences 112 (1998) 39–49.

[13] A. Kusiak, Decomposition in data mining: An industrial case study, IEEE Transaction on Electronics Packaging Manufacturing 23 (4) (2000) 345–353.

[14] A. Kusiak, Rough set theory: A data mining tool for semiconductor manufacturing, IEEE Transaction on Electronics Packaging Manufacturing 24 (1) (2001) 44–50.

[15] T. Li, J. Ma, Y. Xu, N. Yang, An approach to attribute generalization in incomplete information system, in: International Conference on Machine Learning and Cybernetics, 2003, pp. 1678–1691.

[16] T. Li, Y. Xu, A generalization rough set approach to attribute generalization in data mining, Journal of Southwest Jiaotong University 8 (1) (2000) 69–75.

[17] T. Li, N. Yang, Y. Xu, J. Ma, An incremental algorithm for mining classification rules in incomplete information system, in: International Conference of the North American Fuzzy Information, 2004, pp. 446–449.

[18] P.J. Lingras, Y.Y. Yao, Data mining using extensions of the rough set model, Journal of America Society for Information Sciences l49 (1998) 415–422.

[19] Q. Liu, Z. Huang, S. Liu, L. Yao, Decision rules with rough operator and soft computing of data mining, Journal of Computer Research & Development 36 (7) (1999) 800–804.

[20] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning about Data, Kluwer, Dordrecht, 1991.

[21] Z. Pawlak, Rough sets, Communications of the ACM 38 (11) (1995) 89–95.

[22] R.K. Pearson, The problem of disguised missing data, SIGKDD Explorations 8 (1) (2006) 83–92.

[23] J.F. Peters, Z. Suraj, S. Shan, S. Ramanna, W. Pedrycz, N. Pizzi, Classification of meteorological volumetric radar data using rough set methods, Pattern Recognition Letters 24 (6) (2003) 911–920.

[24] L. Polkowski, T.Y. Lin, S. Tsumoto (Eds.), Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems, Physica-Verlag, Heidelberg, 2000.

[25] A.M. Radzikowska, E.E. Kerre, A comparative study of fuzzy rough sets, Fuzzy Sets and Systems 126 (2002) 137–156.

[26] C. Shang, Q. Shen, Aiding classification of gene expression data with feature selection: A comparative study, International Journal of Computational Intelligence Research 1 (1) (2005) 68–76.

[27] J. Stefanowski, A. Tsoukias, On the extension of rough sets under incomplete information, Lecture Notes in Artificial Intelligence 1711 (1999) 73–81.

[28] J. Stefanowski, A. Tsoukias, Incomplete information tables and rough classification, Computational Intelligence 17 (2001) 545–566.

[29] S. Tsumoto, Mining diagnostic rules from clinical databases using rough sets and medical diagnostic model, Information Sciences 162 (2) (2004) 65–80.

[30] J.T. Yao, M. Zhang, Feature selection with adjustable criteria, Lecture Notes in Artificial Intelligence 3641 (2005) 204–213.

[31] Z. Zheng, G. Wang, RRIA: A rough set and rule tree based incremental knowledge acquisition algorithm, Fundamenta Informaticae 59 (2–3) (2004) 299–313.