Int. J. Appl. Math. Comput. Sci., 2001, Vol.11, No.3, 603–619

A ROUGH SET-BASED KNOWLEDGE DISCOVERY PROCESS

Ning ZHONG*, Andrzej SKOWRON**

The knowledge discovery from real-life databases is a multi-phase process consisting of numerous steps, including attribute selection, discretization of real-valued attributes, and rule induction. In the paper, we discuss a rule discovery process that is based on rough set theory. The core of the process is a soft hybrid induction system called the Generalized Distribution Table and Rough Set System (GDT-RS) for discovering classification rules from databases with uncertain and incomplete data. The system is based on a combination of Generalization Distribution Table (GDT) and the Rough Set methodologies. In the preprocessing, two modules, i.e. Rough Sets with Heuristics (RSH) and Rough Sets with Boolean Reasoning (RSBR), are used for attribute selection and discretization of real-valued attributes, respectively. We use a slope-collapse database as an example showing how rules can be discovered from a large, real-life database.

Keywords: rough sets, KDD process, hybrid systems

1. Introduction

The Knowledge Discovery from Databases (KDD) is usually a *multi-phase* process involving numerous steps, like data preparation, preprocessing, search for hypothesis generation, pattern formation, knowledge evaluation, representation, refinement and management. Furthermore, the process may be repeated at different stages when a database is updated (Fayyad *et al.*, 1996).

The *multi-phase* process is an important methodology for the knowledge discovery from real-life data (Zhong *et al.*, 1997). Although the process-centric view has recently been widely accepted by researchers in the KDD community, few KDD systems provide capabilities that a more complete process should possess.

Rough set theory constitutes a sound basis for KDD. It offers useful tools for discovering patterns hidden in data in many aspects (Lin and Cercone, 1997; Pal and Skowron, 1999; Pawlak, 1982; 1991; Skowron and Rauszer, 1992). It can be used in different phases of the knowledge discovery process, like attribute selection, attribute extraction, data reduction, decision rule generation and pattern extraction (templates,

* Department of Information Engineering, Maebashi Institute of Technology, 460–1, Kamisadori-Cho, Maebashi-City, 371, Japan, e-mail: zhong@maebashi-it.ac.jp

** Institute of Mathematics, Warsaw University, ul. Banacha 2, 02–097 Warsaw, Poland, e-mail: skowron@mimuw.edu.pl

association rules) (Komorowski *et al.*, 1999). Furthermore, recent extensions of rough set theory (rough mereology) have brought new methods of decomposition of large data sets, data mining in distributed and multi-agent based environments and granular computing (Polkowski and Skowron, 1996; Polkowski and Skowron, 1999; Yao and Zhong, 1999; Zhong *et al.*, 1999).

In the paper, we discuss a rule discovery process that is based on the rough set approach. In a sense, the rule discovery process described in this paper can be regarded as a demonstration of the process-centered KDD methodology and applications of rough set theory in this process. Section 2 describes a soft hybrid induction system GDT-RS constituting the core in the discovery of classification rules from databases with uncertain and incomplete data. The system is based on a combination of the Generalization Distribution Table (GDT) and the Rough Set methodology. Furthermore, in Sections 3 and 4 we introduce two systems: Rough Sets with Heuristics (RSH) for attribute selection and Rough Sets with Boolean Reasoning (RSBR) for discretization of real-valued attributes, respectively. They are responsible for two steps in the preprocessing realized before the GDT-RS starts. Then, in Section 5, we present an illustrative example of the application of our system for discovering rules from a large, real-life slope-collapse database. Finally, Section 6 gives conclusions and outlines further research directions.

2. Generalized Distribution Table and Rough Set System (GDT-RS)

GDT-RS is a soft hybrid induction system for discovering classification rules from databases with uncertain and incomplete data (Zhong *et al.*, 1998; Dong *et al.*, 1999a). The system is based on a hybridization of the *Generalization Distribution Table (GDT)* and the *Rough Set* methodology. The GDT-RS system can generate, from noisy and incomplete training data, a set of rules with the minimal (semi-minimal) description length, having large strength and covering all instances.

2.1. Generalization Distribution Table (GDT)

We distinguish two kinds of attributes, namely *condition* attributes and *decision* attributes (sometimes called class attributes) in a database. The condition attributes are used to describe possible instances in GDT, while the decision attributes correspond to concepts (classes) described in a rule. Usually, a single decision attribute is all what is required.

Any GDT consists of three components: *possible instances*, *possible generalizations* of instances, and *probabilistic relationships* between possible instances and possible generalizations.

Possible instances, represented at the top row of GDT, are defined by all possible combinations of attribute values from a database. *Possible generalizations* of instances, represented by the left column of a GDT, are all possible cases of generalization

for all possible instances. A wild card '$*$' denotes the generalization for instances[1]. For example, the generalization $*b_0c_0$ means that the attribute a is superfluous (irrelevant) for the concept description. In other words, if an attribute a takes values from $\{a_0, a_1\}$ and both $a_0b_0c_0$ and $a_1b_0c_0$ describe the same concept, the attribute a is superfluous, i.e. the concept can be described by b_0c_0. Therefore, we use the generalization $*b_0c_0$ to describe the set $\{a_0b_0c_0,\ a_1b_0c_0\}$.

The *probabilistic relationships* between possible instances and possible generalizations, represented by entries G_{ij} of a given GDT, are defined by means of a probabilistic distribution describing the strength of the relationship between any possible instance and any possible generalization. The prior distribution is assumed to be uniform if background knowledge is not available[2]. Thus, it is defined by

$$\begin{aligned} G_{ij} &= p(PI_j|PG_i) \\ &= \begin{cases} \dfrac{1}{N_{PG_i}} & \text{if } PG_i \text{ is a generalization of } PI_j, \\ 0 & \text{otherwise,} \end{cases} \end{aligned} \tag{1}$$

where PI_j is the j-th possible instance, PG_i is the i-th possible generalization, and N_{PG_i} is the number of the possible instances satisfying the i-th possible generalization, i.e.

$$N_{PG_i} = \prod_{k\in\{l|\ PG_i[l]=*\}} n_k, \tag{2}$$

where $PG_i[l]$ is the value of the l-th attribute in the possible generalization PG_i, and n_k is the number of values of the k-th attribute. Certainly, we have $\sum_j G_{ij} = 1$ for any i.

Assuming $E = \prod_{k=1}^{m} n_k$, (1) can be rewritten in the following form:

$$\begin{aligned} G_{ij} &= p(PI_j|PG_i) \\ &= \begin{cases} \dfrac{\displaystyle\prod_{k\in\{l|\ PG_i[l]\neq *\}} n_k}{E} & \text{if } PG_i \text{ is a generalization of } PI_j, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{3}$$

Furthermore, the rule discovery can be constrained by three types of biases corresponding to three components of the GDT, so that the user can select more general concept descriptions from an upper level or more specific ones from a lower level, adjust the strength of the relationship between instances and their generalizations, and define/select possible instances (Zhong *et al.*, 1998).

[1] For simplicity, the wild card will sometimes be omitted in the paper.

[2] How to use background knowledge in the rule discovery process is not discussed here due to the limitation on the paper volume. For such a discussion, see the paper (Zhong *et al.*, 2000).

2.2. Rule Strength

Let us recall some basic notions regarding rule discovery from databases represented by decision tables (Komorowski *et al.*, 1999). A decision table (DT) is the quadruple $T = (U, A, C, D)$, where U is a nonempty finite set of objects called the universe, A is a nonempty finite set of primitive attributes, and $C, D \subseteq A$ are two subsets of attributes that are called the condition and decision attributes, respectively (Pawlak, 1991; Skowron and Rauszer, 1992). By $IND(B)$ we denote the indiscernibility relation defined by $B \subseteq A$, $[x]_{IND(B)}$ denotes the indiscernibility (equivalence) class defined by x, and U/B denotes the set of all indiscernibility classes of $IND(B)$. A descriptor over $B \subseteq A$ is any pair (a, v) where $a \in A$ and v is a value of a. If P is a conjunction of some descriptors over $B \subseteq A$, then we denote by $[P]_B$ (or $[P]$) the set of all the objects in DT satisfying P.

In our approach, the rules are expressed in the following form:

$$P \to Q \text{ with } S,$$

i.e. '**if** P **then** Q with strength S', where P denotes a conjunction of descriptors over C (with non-empty set $[P]_{DT}$), Q denotes a concept that the rule describes, and S is a 'measure of the strength' of the rule, defined by

$$S(P \to Q) = s(P) \times (1 - r(P \to Q)), \tag{4}$$

where $s(P)$ is the strength of the generalization P (i.e. the condition of the rule) and r is the noise rate function. The strength of a given rule reflects the incompleteness and uncertainty in the process of rule inducing influenced by both unseen instances and noise.

On the assumption that the prior distribution is uniform, the strength of the generalization $P = PG$ is given by

$$s(P) = \sum_l p(PI_l|P) = \frac{1}{N_P} \text{card}\,([P]_{DT}), \tag{5}$$

where card $([P]_{DT})$ is the number of the observed instances satisfying the generalization P. The strength of the generalization P represents explicitly the prediction for unseen instances. On the other hand, the noise rate is given by

$$r(P \to Q) = 1 - \frac{\text{card}([P]_{DT} \cap [Q]_{DT})}{\text{card}([P]_{DT})}. \tag{6}$$

It shows the quality of classification measured by the number of the instances satisfying the generalization P which cannot be classified into class Q. The user can specify an allowed noise level as a threshold value. Thus, the rule candidates with a noise level larger than the given threshold value will be deleted.

One can observe that the rule strength we propose is equal to its confidence (Agrawal *et al.*, 1996) modified by the strength of the generalization appearing on the left-hand side of the rule. The reader can find in the literature other criteria for rule strength estimation (Bazan, 1998; Grzymała-Busse, 1998; Mitchell, 1997).

2.3. Simplification of the Decision Table by GDT-RS

The process of rule discovery consists of the decision table preprocessing, including selection and extraction of the relevant attributes (features), and the appropriate decision rule generation. The relevant decision rules can be induced from the minimal rules (i.e. with the minimal length of their left-hand sides with respect to the discernibility between decisions) by tuning them (e.g. dropping some conditions to obtain more general rules which are better predisposed to classify new objects even if they do not classify properly some objects from the training set). The relevant rules can be induced from the set of all minimal rules, or from its subset covering the set of objects of a given decision table (Komorowski *et al.*, 1999; Pawlak and Skowron, 1993). A representative approach to the problem of generation of the so-called local relative reducts of condition attributes is the one to represent knowledge to be preserved about the discernibility between objects by means of the discernibility functions (Pawlak, 1991; Skowron and Rauszer, 1992).

It is obvious that by using the GDT one instance can be matched by several possible generalizations, and several instances can be generalized into one possible generalization. Simplifying a decision table by means of the GDT-RS system leads to a minimal (or sub-minimal) set of generalizations covering all instances. The main goal is to find a relevant (i.e. minimal or semi-minimal with respect to the description size) covering of instances still allowing us to resolve conflicts between different decision rules recognizing new objects. The first step in the GDT-RS system for decision rule generation is based on computing local relative reducts of condition attributes by means of the discernibility matrix method (Bazan and Szczuka, 2000; Pawlak, 1991; Skowron and Rauszer, 1992).

Moreover, instead of searching for dispensable attributes, we are rather searching for relevant attributes using a bottom-up method. Any generalization matching instances with different decisions should be checked by means of (6). If the noise level is smaller than a threshold value, such a generalization is regarded as a reasonable one. Otherwise, the generalization is contradictory.

Furthermore, a rule in the GDT-RS is selected according to its priority. The priority can be defined by the number of instances covered (matched) by a rule (i.e. the more instances are covered, the higher the priority is), by the number of attributes occurring on the left-hand side of the rule (i.e. the fewer attributes, the higher the priority is), or by the rule strength (Zhong *et al.*, 1998).

2.4. Searching Algorithm for an Optimal Set of Rules

We now outline the idea of a searching algorithm for a set of rules developed in (Dong *et al.*, 1999a) and based on the GDT-RS methodology. We use a sample decision table shown in Table 1 to illustrate the idea. Let T_{noise} be a threshold value.

Step 1. Create the GDT.

If prior background knowledge is not available, the prior distribution of a generalization is calculated using eqns. (1) and (2).

Table 1. A sample database.

U \ A	a	b	c	d
u_1	a_0	b_0	c_1	y
u_2	a_0	b_1	c_1	y
u_3	a_0	b_0	c_1	y
u_4	a_1	b_1	c_0	n
u_5	a_0	b_0	c_1	n
u_6	a_0	b_2	c_1	n
u_7	a_1	b_1	c_1	y

Step 2. Consider the indiscernibility classes with respect to the condition attribute set C (such as u_1, u_3 and u_5 in the sample database of Table 1) as one instance, called the *compound instance* (such as $u_1^{'} = [u_1]_{IND(a,b,c)}$ in the following table). Then the probabilities of generalizations can be calculated correctly.

U \ A	a	b	c	d
$u_1^{'}, (u_1, u_3, u_5)$	a_0	b_0	c_1	y, y, n
u_2	a_0	b_1	c_1	y
u_4	a_1	b_1	c_0	n
u_6	a_0	b_2	c_1	n
u_7	a_1	b_1	c_1	y

Step 3. For any compound instance u' (such as the instance $u_1^{'}$ in the above table), let $d(u')$ be the set of the decision classes to which the instances in u' belong. Furthermore, let $X_v = \{x \in U : d(x) = v\}$ be the decision class corresponding to the decision value v. The rate r_v can be calculated by (6). If there exists a $v \in d(u')$ such that $r_v(u') = \min\{r_{v'}(u') | v' \in d(u')\} < T_{\text{noise}}$, then we let the compound instance u' point to the decision class corresponding to v. If there is no $v \in d(u')$ such that $r_v(u') < T_{\text{noise}}$, we treat the compound instance u' as a contradictory one, and set the decision class of u' to $\perp(uncertain)$. For example, we have

U \ A	a	b	c	d
$u_1^{'}(u_1, u_3, u_5)$	a_0	b_0	c_1	$\perp$

Let $U^{'}$ be the set of all the instances except the contradictory ones.

Step 4. Select one instance u from $U^{'}$. Using the idea of the discernibility matrix, create a discernibility vector (i.e. the row or the column with respect to u in the discernibility matrix) for u. For example, the discernibility vector for instance $u_2 : a_0 b_1 c_1$ is as follows:

U \ U	$u_1^{'}(\perp)$	$u_2(y)$	$u_4(n)$	$u_6(n)$	$u_7(y)$
$u_2(y)$	b	$\emptyset$	a, c	b	$\emptyset$

Step 5. Compute all the so-called local relative reducts for instance u by using the discernibility function. For example, from instance $u_2 : a_0b_1c_1$, we obtain two reducts, $\{a, b\}$ and $\{b, c\}$:

$$f_T(u_2) = (b) \wedge \top \wedge (a \vee c) \wedge (b) \wedge \top \; = \; (a \wedge b) \vee (b \wedge c).$$

Step 6. Construct rules from the local reducts for instance u, and revise the strength of each rule using (4). For example, the following rules are acquired:

$$\{a_0b_1\} \rightarrow y \;\text{ with }\; S = 1 \times \frac{1}{2} = 0.5, \text{ and}$$

$$\{b_1c_1\} \rightarrow y \;\text{ with }\; S = 2 \times \frac{1}{2} = 1$$

for instance $u_2 : a_0b_1c_1$.

Step 7. Select the best rules from the rules (for u) obtained in *Step 6* according to its priority (Zhong *et al.*, 1998). For example, the rule '$\{b_1c_1\} \rightarrow y$' is selected for the instance $u_2 : a_0b_1c_1$ because it matches more instances than the rule '$\{a_0b_1\} \rightarrow y$'.

Step 8. $U' = U' - \{u\}$. If $U' \neq \emptyset$, then go back to *Step 4*. Otherwise, go to *Step 9*.

Step 9. If any rule selected in *Step 7* covers exactly one instance, then STOP, otherwise, using the method from Section 2.3, select a minimal set of rules covering all instances in the decision table.

The following table shows the result for the sample database shown in Table 1:

U	rules	strengths
u_2, u_7	$b_1 \wedge c_1 \rightarrow y$	1
u_4	$c_0 \rightarrow n$	0.167
u_6	$b_2 \rightarrow n$	0.25

The time complexity of the algorithm is $O(mn^2Nr_{\max})$, where n is the number of instances in a given database, m stands for the number of attributes, $Nr_{\max}$ is the maximal number of reducts for instances.

One can see that the algorithm is not suitable for databases with large numbers of attributes or reducts. A possible way of settling the issue is to use another algorithm called the *Sub-Optimal Solution*, which is more suitable for such databases (Dong *et al.*, 1999a). Another method to solving the problem is to find a reduct (subset) of condition attributes in preprocessing before the algorithm of (Dong *et al.*, 1999b) is used. We describe such a method in the following section.

3. Rough Sets with Heuristics (RSH)

RSH is a system for an attribute subset selection. It is based on rough sets with heuristics (Dong *et al.*, 1999b). The development of the RSH is based on the following observations: (i) a database always contains a lot of attributes that are redundant and

not necessary for rule discovery; (ii) if these redundant attributes are not removed, not only does the time complexity of the rule discovery increase, but also the quality of the discovered rules can be significantly decreased.

The goal of attribute selection is to find an optimal subset of attributes according to some criterion so that a classifier with the highest possible accuracy can be induced by an inductive learning algorithm using information about data available only from the subset of attributes.

3.1. Rough Sets with Heuristics

In this section we explain some concepts of rough sets related to attribute selection in preprocessing (Pawlak, 1991). Let C and D denote the condition and decision attribute sets of the decision table T, respectively. The C-positive region of D is the set of all objects from the universe U which can be classified with certainty to classes of U/D employing attributes from C, i.e.

$$POS_C(D) = \bigcup_{X \in U/D} \underline{C}X,$$

where $\underline{C}X$ denotes the *lower approximation* of the set X with respect to C, i.e. the set of all objects from U that can be classified with certainty as elements of X based on attributes from C.

An attribute c ($c \in C$) is *dispensable* in a decision table T, if $POS_{(C-\{c\})}(D) = POS_C(D)$; otherwise the attribute c is *indispensable* in T. A set of attributes $R \subseteq C$ is called a *reduct* of C if it is a minimal attribute subset preserving the condition $POS_R(D) = POS_C(D)$. Furthermore, the set of all the attributes indispensable in C is denoted by $CORE(C)$. We have

$$CORE(C) = \bigcap RED(C),$$

where $RED(C)$ is the set of all the reducts of C.

The quality of an attribute subset R in the GDT-RS depends on the strength of the rules discovered by using this subset. The higher the strength, the better the subset is. Searching for attributes that are of benefit to acquire rules with large cover rate and strength is based on the selection strategy described in the following section.

3.2. Heuristic Algorithm for Feature Selection

We use the attributes from $CORE$ as an initial attribute subset. Next, we select attributes one by one from among the unselected ones using some strategies, and we add them to the attribute subset until a reduct approximation is obtained.

Algorithm:
Let R be a set of selected condition attributes, P a set of unselected condition attributes, U a set of all instances, and $EXPECT$ an accuracy threshold. In the initial state, we set $R = CORE(C)$, $P = C - CORE(C)$, $k = 0$.

Table 2. Another sample database.

$U \setminus A$	a	b	c	d	e
u_1	a_1	b_0	c_2	d_1	e_1
u_2	a_1	b_0	c_2	d_0	e_1
u_3	a_1	b_2	c_0	d_0	e_2
u_4	a_1	b_2	c_2	d_1	e_0
u_5	a_2	b_1	c_0	d_0	e_2
u_6	a_2	b_1	c_1	d_0	e_2
u_7	a_2	b_1	c_2	d_1	e_1

Step 1. Remove all consistent instances: $U = U - POS_R(D)$.

Step 2. **If** $k \geq EXPECT$, where

$$k = \gamma_R(D) = \frac{\text{card}\,(POS_R(D))}{\text{card}\,(U)}, \ \textbf{then}\ STOP$$

else if $POS_R(D) = POS_C(D)$, return *'only* $k = \text{card}\,(POS_C(D))/\text{card}\,(U)$ *is available'* and $STOP$.

Step 3. Calculate

$$v_p = \text{card}\,(POS_{R\cup\{p\}}(D)),$$
$$m_p = \text{max\_size}\,(POS_{(R\cup\{p\})}(D))/(R \cup \{p\} \cup D) \text{ for any } p \in P.$$

Step 4. Choose the best attribute p, i.e. that with the largest $v_p \times m_p$, and set

$$R = R \cup \{p\},\ P = P - \{p\};$$

Step 5. Go back to *Step 2.*

Illustrative Example. We select an attribute subset using the above algorithm for the sample database shown in Table 2. Here a, b, c and d are condition attributes, e stands for the decision attribute, $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7\}$, b is the unique indispensable attribute (deleting b will cause an inconsistency: $\{a_1c_2d_1\} \to e_1$ and $\{a_1c_2d_1\} \to e_0$).

From the families of equivalence classes $U/\{b\} = \{\{u_1, u_2\},\ \{u_5, u_6, u_7\},\ \{u_3, u_4\}\}$ and $U/\{e\} = \{\{u_4\}, \{u_1, u_2, u_7\},\ \{u_3, u_5, u_6\}\}$, we obtain the $\{b\}$-*positive region* of $\{e\}$: $POS_{\{b\}}(\{e\}) = \{u_1, u_2\}$. Hence, in the initial state we have $R = \{b\}$, $P = \{a, c, d\}$ and $U = \{u_3, u_4, u_5, u_6, u_7\}$. The initial state is shown in Table 3.

Setting $EXPECT = 1$, the termination condition will be $k \geq 1$. Since $k = 2/7 < 1$, R is not a reduct, and we must continue to select condition attributes. The next

Table 3. The initial state for attribute selection.

$U \setminus A$	b	e
u_3	b_2	e_2
u_4	b_2	e_0
u_5	b_1	e_2
u_6	b_1	e_2
u_7	b_1	e_1

Table 4. Selecting the second attribute from $R = \{a, c, d\}$.

$U \setminus A$	a	b	e
u_3	a_1	b_2	e_2
u_4	a_1	b_2	e_0
u_5	a_2	b_1	e_2
u_6	a_2	b_1	e_2
u_7	a_2	b_1	e_1

1. Selecting $\{a\}$

$U \setminus A$	b	c	e
u_3	b_2	c_0	e_2
u_4	b_2	c_2	e_0
u_5	b_1	c_0	e_2
u_6	b_1	c_1	e_2
u_7	b_1	c_2	e_1

2. Selecting $\{c\}$

$U \setminus A$	b	d	e
u_3	b_2	d_0	e_2
u_4	b_2	d_1	e_0
u_5	b_1	d_0	e_2
u_6	b_1	d_0	e_2
u_7	b_1	d_1	e_1

3. Selecting $\{d\}$

candidates are a, c or d. Table 4 gives the results of adding $\{a\}$, $\{c\}$, and $\{d\}$ to R, respectively.

From Table 4 we obtain the following families of equivalence classes:

$$U/\{e\} = \{\{u_3, u_5, u_6\}, \{u_4\}, \{u_7\}\},$$

$$U/\{a, b\} = \{\{u_3, u_4\}, \{u_5, u_6, u_7\}\},$$

$$U/\{b, c\} = \{\{u_3\}, \{u_4\}, \{u_5\}, \{u_6\}, \{u_7\}\},$$

$$U/\{b, d\} = \{\{u_3\}, \{u_4\}, \{u_5, u_6\}, \{u_7\}\}.$$

We also have

$$POS_{\{a,b\}}(\{e\}) = \emptyset,$$

$$POS_{\{b,c\}}(\{e\}) = POS_{\{b,d\}}(\{e\}) = \{u_3, u_4, u_5, u_6, u_7\},$$

$$\max\_\text{size}(POS_{\{b,c\}}(\{e\})/\{b, c, e\}) = 1,$$

$$\max\_\text{size}(POS_{\{b,d\}}(\{e\})/\{b, d, e\}) = \text{card}\,(\{u5, u6\}) = 2.$$

One can see that by selecting the attribute a we cannot reduce the number of contradictory instances, but if either c or d is chosen, then all instances become consistent. Since the maximal set is in $U/\{b, d, e\}$, then, according to our selection strategies, d should be selected first.

After adding d to R, all instances are consistent and must be removed from U. Hence U becomes empty, $k = 1$, and the process is finished. Thus, the selected attribute subset is $\{b, d\}$. ♦

4. Rough Sets and Boolean Reasoning (RSBR)

RSBR is a system for discretization of real-valued attributes. Discretization of real-valued attributes is an important preprocessing step in our rule discovery process. The development of RSBR is based on the following observations: (i) real-life data sets often contain mixed types of data such as real-valued, symbolic data, etc.; (ii) real-valued attributes should be discretized in preprocessing; (iii) the choice of the discretization method depends on the analyzed data.

The core module in our rule discovery process is the GDT-RS. In the GDT-RS, the probabilistic distribution between possible instances and possible generalizations depends on the number of the values of attributes. The rules induced without discretization are of low quality because they will usually not recognize new objects.

4.1. Discretization Based on RSBR

In order to solve the discretization problems, we have developed a discretization system called the RSBR that is based on hybridization of rough sets and Boolean reasoning proposed in (Nguyen and Skowron, 1995; Nguyen and Skowron, 1997).

A great effort has been made (Fayyad and Irani, 1992; Chmielewski and Grzymała-Busse, 1994; Dougherty *et al.*, 1995; Nguyen and Nguyen, 1998) to find effective methods of discretization of real-valued attributes. We may obtain different results by using different discretization methods. The results of discretization affect directly the quality of the discovered rules. Some of discretization methods totally ignore the effect of the discretized attribute values on the performance of the induction algorithm. The RSBR combines discretization of real-valued attributes and classification. In the process of the discretization of real-valued attributes we should also take into account the effect of the discretization on the performance of our induction system GDT-RS.

Roughly speaking, the basic concepts of the discretization based on the RSBR can be summarized as follows: (i) discretization of a decision table, where $V_c = [v_c, w_c)$ is an interval of real values taken by attribute c, is a searching process for a partition P_c of V_c for any $c \in C$ satisfying some optimization criteria (like a minimal partition) while preserving some discernibility constraints (Nguyen and Skowron, 1995; Nguyen and Skowron, 1997); (ii) any partition of V_c is defined by a sequence of the so-called *cuts* $v_1 < v_2 < \cdots < v_k$ from V_c; (iii) any family of partitions $\{P_c\}_{c \in C}$ can be identified with a set of cuts.

Table 5 shows an example of discretization. The discretization process returns a partition of the value sets of condition attributes into intervals:

$$P = \{(a, 0.9), (a, 1.5), (b, 0.75), (b, 1.5)\}.$$

4.2. Algorithm

The main steps of our algorithm can be described as follows:

Step 1. Define a set of Boolean variables $BV(U)$. For the example shown in Table 5 we have $BV(U) = \{p_1^a, p_2^a, p_3^a, p_4^a, p_1^b, p_2^b, p_3^b\}$, where p_1^a corresponds to the inter-

Table 5. An example of discretization.

U	a	b	d
x_1	0.8	2	1
x_2	1	0.5	0
x_3	1.3	3	0
x_4	1.4	1	1
x_5	1.4	2	0
x_6	1.6	3	1
x_7	1.3	1	1

$\Longrightarrow$

U	a^p	b^p	d
x_1	0	2	1
x_2	1	0	0
x_3	1	2	0
x_4	1	1	1
x_5	1	2	0
x_6	2	2	1
x_7	1	1	1

val [0.8, 1) of a; p_2^a corresponds to the interval [1,1.3) of a; p_3^a corresponds to the interval [1.3, 1.4) of a; p_4^a corresponds to the interval [1.4, 1.6) of a; p_1^b corresponds to the interval [0.5, 1) of b; p_2^b corresponds to the interval [1, 2) of b; p_3^b corresponds to the interval [2, 3) of b.

Step 2. Create a new decision table T^p by using the set of Boolean variables defined in *Step* 1. Here T^p is called the *P-discretization of* T, $T^p = (U, \cup\{d\}, A^p, d)$, p_k^c is a propositional variable corresponding to the interval $[v_k^c, v_{k+1}^c)$ for any $k \in \{1, \ldots, n_c - 1\}$ and $c \in C$.

Table 6 shows an example of T^p. We set, e.g. $p_1^a(x_1, x_2) = 1$, because any cut in the interval $[0.8, 1)$ corresponding to p_1^a discerns x_1 and x_2.

Step 3. Find a minimal subset of P that discerns all the objects in different decision classes by using the discernibility formula

$$\Phi^U = \wedge\{\psi(i,j) : d(x_i) \neq d(x_j)\},$$

where, e.g. $\psi(i,j) = p_1^a \vee p_1^b \vee \vee p_2^b$ means that in order to discern object x_1 and x_2, at least one of the following cuts must be selected: (i) a cut between $a(0.8)$ and $a(1)$; (ii) a cut between $b(0.5)$ and $b(1)$; (iii) a cut between $b(1)$ and $b(2)$.

From Table 6 we obtain the discernibility formula

$$\begin{aligned}
\Phi^U = {} & (p_1^a \vee p_1^b \vee p_2^b) \wedge (p_1^a \vee p_2^a \vee p_3^b) \\
& \wedge\ (p_1^a \vee p_2^a \vee p_3^a) \\
& \wedge\ (p_2^a \vee p_3^a \vee p_1^b) \wedge (p_2^a \vee p_2^b \vee p_3^b) \\
& \wedge\ (p_2^a \vee p_3^a \vee p_4^a \vee p_1^b \vee p_2^b \vee p_3^b) \\
& \wedge\ (p_3^a \vee p_4^a) \wedge (p_4^a \vee p_3^b) \wedge (p_2^a \vee p_1^b) \\
& \wedge\ (p_2^b \vee p_3^b) \wedge (p_3^a \vee p_2^b) \wedge p_2^b.
\end{aligned}$$

Table 6. An example of T^p.

U^*	p_1^a	p_2^a	p_3^a	p_4^a	p_1^b	p_2^b	p_3^b
(x_1, x_2)	1	0	0	0	1	1	0
(x_1, x_3)	1	1	0	0	0	0	1
(x_1, x_5)	1	1	1	0	0	0	0
(x_4, x_2)	0	1	1	0	1	0	0
(x_4, x_3)	0	0	1	0	0	1	1
(x_4, x_5)	0	0	0	0	0	1	0
(x_6, x_2)	0	1	1	1	1	1	1
(x_6, x_3)	0	0	1	1	0	0	0
(x_6, x_5)	0	0	0	1	0	0	1
(x_7, x_2)	0	1	0	0	1	0	0
(x_7, x_3)	0	0	0	0	0	1	1
(x_7, x_5)	0	0	1	0	0	1	0

Finally, we obtain four prime implicants denoted by the discernibility formula in DNF form,

$$\Phi^U = (p_2^a \wedge p_4^a \wedge p_2^b) \vee (p_2^a \wedge p_3^a \wedge p_2^b) \wedge p_3^b)$$
$$\vee\ (p_3^a \wedge p_1^b \wedge p_2^b \wedge p_3^b) \vee (p_1^a \wedge p_4^a \wedge p_1^b \wedge p_2^b).$$

Furthermore, we select $\{p_2^a, p_4^a, p_2^b\}$, i.e. $P = \{(a, 1.2), (a, 1.5), (b, 1.5)\}$ as the optimal result, because it is the minimal subset of P preserving discernibility.

5. Application

We use a slope-collapse database as an example. The slope-collapse database consists of data of the dangerous natural steep slopes in the Yamaguchi region, Japan. There are 3436 instances in this database. Among them 430 places were collapsed, and 3006 were not. There are 32 condition attributes and 1 decision attribute. The task is to find the reason that causes the slope to collapse.

The attributes are listed in Table 7, where *collapse* is a decision attribute and the remaining 32 attributes are condition attributes. Eight attributes such as 'collapsing history of current slope', 'collapsing history of adjacent slope', 'no. of active fault', 'countermeasure work', etc. are obviously irrelevant for the rule discovery. They are removed before attribute selection. From the remaining 24 condition attributes, 9 attributes were selected by using RSH (see Table 8).

The rule discovery on the data set restricted to the selected attributes was realized by using the GDT-RS. Table 9 shows conditions causing the slope to collapse. We list only examples of rules with higher strength. In the table, *Used* denotes the number of instances covered by the rule, *Strength* indicates the strengths of the generalization (conditions), which can be calculated from (5). Here $E = \prod_{i=1}^{m} n_i$, where n_i is the number of values of the i-th condition attribute, n = [2, 27, 9, 9, 10, 5, 5, 2, 6, 3]. The real-valued attributes were discretized using RSBR.

Table 7. The condition attributes in the slope-collapse database.

Attribute name	Number of values
extension of collapsed steep slope	real
gradient	real
altitude	real
slope azimuthal	9
slope shape	9
direction of high rank topography	10
shape of transverse section	5
transition line	3
position of transition line	5
condition of earth surface	5
thickness of soil surface	2
condition of ground	6
condition of base rock	4
relation between slope and unsuccessive face	7
fault, broken region	4
condition of weather	5
kind of plant	6
age of tree	7
condition of lumbering	4
collapsing history of current slope	3
condition of current slope	5
collapsing history of adjacent slope	3
condition of adjacent slope	6
spring water	4
countermeasure work	3
state of upper part of countermeasure work	5
state of upper part of countermeasure work2	6
state of upper part of countermeasure work3	7
No. of active fault	real
active fault traveling	7
distance between slope and active fault	real
direction of slope and active fault	9

The results were evaluated by an expert who did the same work on similar data by using a discriminant analysis. He picked out the important factors (attributes) about the 'collapse' from the same data. The attributes selected by using our approach are almost the same as the most important factors (attributes) selected by the expert.

6. Conclusion

We have presented a rule discovery process based on the rough set approach to discovering *classification* rules in databases. The rule discovery process described in this paper demonstrates the usefulness of rough set theory and is the basic one implemented in the GLS discovery system (Zhong and Ohsuga, 1995; Zhong *et al.*, 1997).

Table 8. The attribute subset selected from the slope-collapse database.

Attribute name	Short name	Number of values
altitude	altitude	real
slope azimuthal	s_azimuthal	9
slope shape	s_shape	9
direction of high rank topography	direction_high	10
shape of transverse section	t_shape	5
position of transition line	tl_position	5
thickness of soil surface	soil_thick	real
kind of plant	plant_kind	6
distance between slope and active fault	s_f_distance	real

Table 9. The results of the slope collapse.

Conditions	Used	Strength
s_azimuthal(2) $\wedge$ s_shape(5) $\wedge$ direction_high(8) $\wedge$ plant_kind(3)	5	(4860/E)
altitude[21,25) $\wedge$ s_azimuthal(3) $\wedge$ soil_thick($\geq$ 45)	5	(486/E)
s_azimuthal(4) $\wedge$ direction_high(4) $\wedge$ t_shape(1) $\wedge$ tl_position(2) $\wedge$ s_f_distance($\geq$ 9)	4	(6750/E)
altitude[16,17) $\wedge$ s_azimuthal(3) $\wedge$ soil_thick($\geq$ 45) $\wedge$ _f_distance($\geq$ 9)	4	(1458/E)
altitude[20,21) $\wedge$ t_shape(3) $\wedge$ tl_position(2) $\wedge$ plant_kind(6) $\wedge$ s_f_distance($\geq$ 9)	4	(12150/E)
altitude[11,12) $\wedge$ s_azimuthal(2) $\wedge$ tl_position(1)	4	(1215/E)
altitude[12,13) $\wedge$ direction_high(9) $\wedge$ tl_position(4) $\wedge$ s_f_distance[8,9)	4	(4050/E)
altitude[12,13) $\wedge$ s_azimuthal(5) $\wedge$ t_shape(5) $\wedge$ s_f_distance[8,9)	4	(3645/E)
altitude[36,37) $\wedge$ plant_kind(5)	3	(162/E)
altitude[13,14) $\wedge$ s_shape(2) $\wedge$ direction_high(4)	3	(2430/E)
altitude[8,9) $\wedge$ s_azimuthal(3) $\wedge$ s_shape(2)	3	(2187/E)
altitude[18,19) $\wedge$ s_shape(4) $\wedge$ plant_kind(2)	3	(1458/E)

The process based on the rough set approach can be further extended by including granular computing, decomposition of large databases, and rule discovery in distributed environments (Yao and Zhong, 1999; Polkowski and Skowron, 1996; Polkowski and Skowron, 1999; Nguyen *et al.*, 1999). Our paper constitutes a first step toward a multi-strategy and multi-agent discovery system.

Acknowledgements

The authors would like to thank Prof. H. Nakamura and Mr. Hiro for providing the slope collapse database and background knowledge, and for evaluating the experimental results. The research of Andrzej Skowron was supported by Grant No. 8T11C 025 19 from the National Committee for Scientific Research (KBN) and by the Wallenberg Foundation.

References

Agrawal R., Mannila H., Srikant R., Toivonen H. and Verkano A. (1996): *Fast discovery of association rules*, In: Advances in Knowledge Discovery and Data Mining (U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, Eds.). — Cambridge, Massachusetts: MIT Press, pp.307–328.

Bazan J.G. (1998): *A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision system*, In: Rough Sets in Knowledge Discovery 1: Methodology and Applications (L. Polkowski, A. Skowron, Eds.). — Heidelberg: Physica-Verlag, pp.321–365.

Bazan J.G. and Szczuka M. (2000): *RSES and RSESlib—A collection of tools for rough set computations*. — Proc. 2nd Int. Conf. *Rough Setsand Current Trends in Computing (RSCTC-2000)*, Banff, pp.74–81.

Chmielewski M.R. and Grzymała-Busse J.W. (1994): *Global discretization of attributes as preprocessing for machine learning*. — Proc. 3rd Int. Workshop *Rough Sets and Soft Computing*, San Tose, pp.294–301.

Dong J.Z., Zhong N. and Ohsuga S. (1999a): *Probabilistic rough induction: The GDT-RS methodology and algorithms*, In: Foundations of Intelligent Systems (Z.W. Ras and A. Skowron, Eds.). — Berlin: Springer, pp.621–629.

Dong J.Z., Zhong N. and Ohsuga S. (1999b): *Using rough sets with heuristics to feature selection*, In: New Directions in Rough Sets, Data Mining, Granular-Soft Computing (N. Zhong, A. Skowron, S. Ohsuga, Eds.). — Berlin: Springer, pp.178–187.

Dougherty J., Kohavi R. and Sahami M. (1995): *Supervised and unsupervised discretization of real features*. — Proc. 12th Int. Conf. *Machine Learning*, pp.194–202.

Fayyad U.M. and Irani K.B. (1992): *On the handling of real-valued attributes in decison tree generation*. — Machine Learning, Vol.8, pp.87–102.

Fayyad U.M., Piatetsky-Shapiro G. and Smyth P. (1996): *From data mining to knowledge discovery: An overview*, In: Advances in Knowledge Discovery and Data Mining (U. Fayyad, G. Piatetsky-Shapiro, Eds.). — Cambridge, Massachusetts: MIT Press, pp.1–36.

Grzymała-Busse J.W. (1998): *Applications of rule induction system LERS*, In: Rough Sets in Knowledge Discovery 1: Methodology and Applications (L. Polkowski, A. Skowron, Eds.). — Heidelberg: Physica-Verlag, pp.366–375.

Komorowski J., Pawlak Z., Polkowski L. and Skowron A. (1999): *Rough sets: A tutorial*, In: Rough Fuzzy Hybridization: A New Trend in Decision Making (S.K. Pal and A. Skowron, Eds.). — Singapore: Springer, pp.3–98.

Lin T.Y. and Cercone N. (Eds.) (1997): *Rough Sets and Data Mining: Analysis of Imprecise Data*. — Boston: Kluwer.

Mitchell T.M. (1997): *Machine Learning*. — Boston: Mc Graw-Hill.

Nguyen H. Son and Skowron A. (1995): *Quantization of real value attributes*. — Proc. Int. Workshop *Rough Sets and Soft Computing* at 2nd Joint Conf. *Information Sciences (JCIS'95)*, Durham, NC, pp.34–37.

Nguyen H. Son and Skowron A. (1997): *Boolean reasoning for feature extraction problems*, In: Foundations of Intelligent Systems (Z.W. Ras, A. Skowron, Eds.). — Berlin: Springer, pp.117–126.

Nguyen H. Son and Nguyen S. Hoa (1998): *Discretization methods in data mining*, In: Rough Sets in Knowledge Discovery (L. Polkowski, A. Skowron, Eds.). — Heidelberg: Physica-Verlag, pp.451–482.

Nguyen S.H., Nguyen H.S. Skowron A. (1999): *Decomposition of task specification problems*, In: Foundations of Intelligent Systems (Z.W. Ras and A. Skowron, Eds.). — Berlin: Springer, pp.310–318.

Pal S.K. and Skowron A. (Eds.) (1999): *Rough Fuzzy Hybridization*. — Singapore: Springer.

Pawlak Z. (1982): *Rough sets*. — Int. J. Comp. Inf. Sci., Vol.11, pp.341–356.

Pawlak Z. (1991): *Rough Sets, Theoretical Aspects of Reasoning about Data*. — Boston: Kluwer.

Pawlak Z. and Skowron A. (1993): *A rough set approach for decision rules generation*. — Proc. Workshop W12: *The Management of Uncertainty in AI* at 13th *IJCAI*, see also: Institute of Computer Science, Warsaw University of Technology, ICS Res. Rep., 23/93, pp.1–19.

Polkowski L. and Skowron A. (1996): *Rough mereology: A new paradigm for approximate reasoning*. — Int. J. Approx. Reasoning, Vol.15, No.4, pp.333–365.

Polkowski L. and Skowron A. (1999): *Towards adaptive calculus of granules*, In: Computing with Words in Information/Intelligent Systems 1: Foundations (L.A. Zadeh and J. Kacprzyk, Eds.). — Heidelberg: Physica-Verlag, pp.201–228.

Skowron A. and Rauszer C. (1992): *The discernibility matrixes and functions in information systems*, In: Intelligent Decision Support (R. Slowinski, Ed.). — Boston: Kluwer, pp.331–362.

Yao Y.Y. and Zhong N. (1999): *Potential Applications of Granular Computing in Knowledge Discovery and Data Mining*. — Proc. 5th Int. Conf. *Information Systems Analysis and Synthesis (IASA'99)*, Orlando, pp.573–580.

Zhong N. and Ohsuga S. (1995): *Toward a multi-strategy and cooperative discovery system*. — Proc. 1st Int. Conf. *Knowledge Discovery and Data Mining (KDD-95)*, Montreal, pp.337–342.

Zhong N., Liu C. and Ohsuga S. (1997): *A way of increasing both autonomy and versatility of a KDD system*, In: Foundations of Intelligent Systems (Z.W. Ras and A. Skowron, Eds.). — Berlin: Springer, pp.94–105.

Zhong N., Dong J.Z. and Ohsuga S. (1998): *Data mining: A probabilistic rough set approach*, In: Rough Sets in Knowledge Discovery, Vol.2 (L. Polkowski and A. Skowron, Eds.). — Heidelberg: Physica-Verlag, pp.127–146.

Zhong N., Skowron A. and Ohsuga S. (Eds.) (1999): *New Directions in Rough Sets, Data Mining, and Granular-Soft Computing*. — Berlin: Springer.

Zhong N., Dong J.Z. and Ohsuga S. (2000): *Using background knowledge as a bias to control the rule discovery process*, In: Principles of Data Mining and Knowledge Discovery (D.A. Zighed, J. Komorowski and J. Zytkow, Eds.). — Berlin: Springer, pp.691–698.