Intelligent Information Systems IV
Proceedings of the Workshop held in
Augustów, Poland, 5-9 June, 1995

pages 62- 73

Decision System Based on Tolerance Rough Sets

Jarosław Stepaniuk, Marek Krętowski

Institute of Computer Science,
Technical University of Bialystok,
Wiejska 45A, 15-351 Bialystok, Poland ;
email: {jstepan,mkret}@ii.pb.bialystok.pl; fax: +4885 422393

Abstract. We present a decision system, which is based on tolerance relations and the rough set theory. It consists from a few almost separate subsystems: searching for tolerance thresholds, data reduction, tolerance decision rules' generation. We investigate various similarity measures and tolerance thresholds to find out tolerance sets. We propose technique whose aim is to reduce the number of examples and the number of attributes involved in the process of learning from examples. We also present algorithm for decision rules' generation and classification of new instances.

Keywords: rough sets, machine learning, knowledge discovery in databases, genetic algorithms

Introduction

Very important feature of each decision system based on the rough set approach [Pa91, Sl92, Zi94] is how it treats cardinal attributes. A lot of systems use preliminary discretization (quantization). We propose another approach based on tolerance relations. We use similarity measures to express differences between objects and values of attributes. We find the optimal values of thresholds to show similarities among examples from one class and differences among classes.

The progress in knowledge discovery [ShLa90, Zy92] from large experimental data sets depends on the development of efficient methods for data reduction. We present a data reduction technique whose aim is to reduce the number of examples and the number of attributes involved in the process of learning from examples.

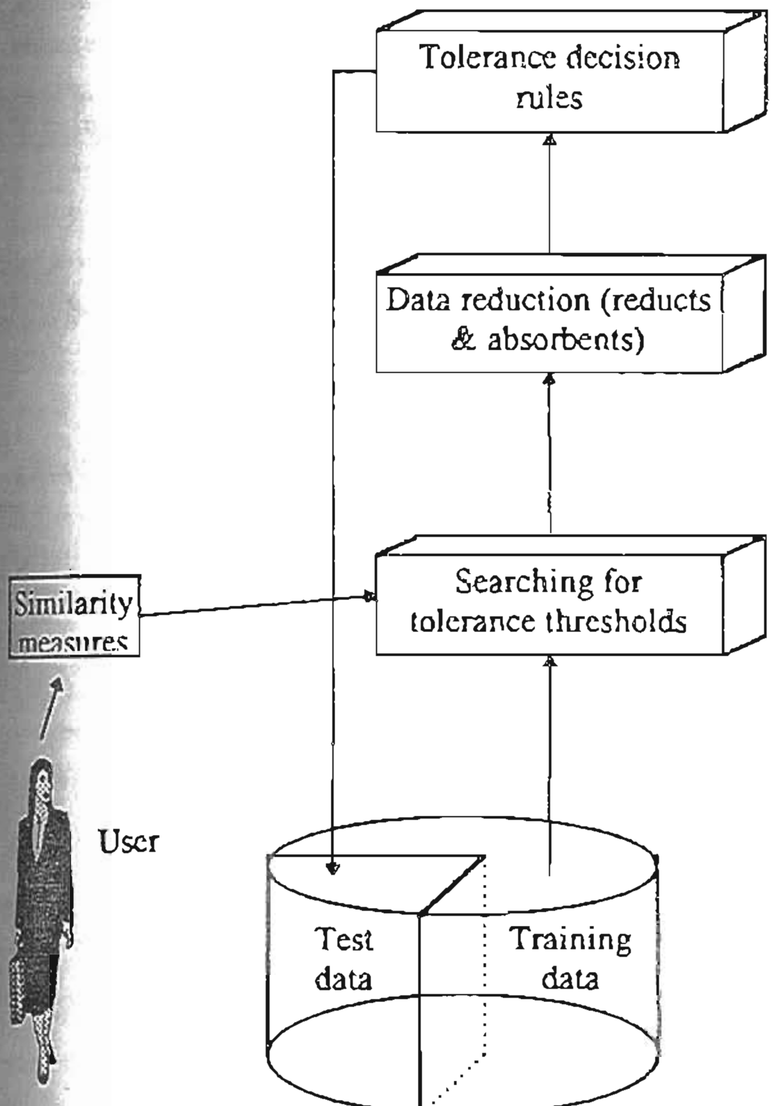

Fig. 1.

Let $\boldsymbol{A} = (U, A \cup \{d\})$ be a decision table [Pa91], where U is a set of objects (examples), A is a set of condition attributes and d is a decision. The reduction process of $\boldsymbol{A}$ consists of finding a new decision table $\boldsymbol{A}' = (U', A' \cup \{d\})$ that satisfies the conditions $U' \subset U$, $A' \subset A$ and the decision rules constructed from $\boldsymbol{A}'$ have (almost) the same quality of classification as the decision rules constructed from $\boldsymbol{A}$. The elements which belong to the new decision table are chosen using an evaluation criterion based on rough set theory [Pa91] and Boolean reasoning [Br90]. More precisely we use the notions of a tolerance attribute reduct [SkSt94] and an absorbent set of object set [Te94, KPSS95]. We also present algorithm for tolerance decision rules' generation and for classifying new objects. General schema of our system is presented at the figure.

This paper is organized as follows. In Section 1 we present some similarity measures between values of a given attribute. In Section 2 we present basic notions concerning rough set concept based on tolerance relations. In Section 3 we describe the problem of finding optimal tolerance thresholds and its efficient solution using genetic algorithm. In Section 4 we investigate some attribute reduction problems for tolerance decision tables. We also consider problem of the number of objects' reduction and we propose a procedure based on the notion of a relative absorbent set. In Section 5 we present some method of decision rules' generation. In Section 6 we include examples of application of the proposed approach.

1 Similarity measures

In this section we present several similarity measures between values of a given attribute.

Let $\mathcal{A} = (U, A \cup \{d\})$ be a decision table and let $r(d)$ be a number of decision values. We define similarity measures between two values of a given attribute $a \in A$. The symbol $n(a)$ will be used to denote $n(a)$-th similarity measure of attribute a.

For attribute $a \in A$with numeric values we define 1-st similarity measure ($n(a) = 1$) by

$$s_a(\nu_i, \nu_j) = 1 - \frac{|\nu_i - \nu_j|}{|a_{\max} - a_{\min}|}$$

where $a_{\min}$, $a_{\max}$ denotes the minimum and maximum values of attribute a, respectively.

Assuming that the values of attribute a are ordered as follows $\nu_1 \leq \nu_2 \leq \ldots \leq \nu_{card(V_a)}$ we let

$$s_a(\nu_i, \nu_j) = 1 - \frac{|i - j|}{card(V_a) - 1} \qquad (n(a) = 2).$$

For attribute a with nominal (categorical) values we consider the following similarity measures

$$s_a(\nu_i, \nu_j) = \begin{cases} 1 & \text{if } \nu_i = \nu_j \\ 0 & \text{if } \nu_i \neq \nu_j \end{cases} \qquad (n(a) = 3)$$

$$s_a(\nu_i, \nu_j) = 1 - \sum_{k=1}^{r(d)} \frac{|P(d = k, a = \nu_i) - P(d = k, a = \nu_j)|}{P(d = k)} \qquad (n(a) = 4)$$

$$s_a(\nu_i, \nu_j) = 1 - \sum_{k=1}^{r(d)} |P(d = k, a = \nu_i) - P(d = k, a = \nu_j)| \qquad (n(a) = 5)$$

$$s_a(\nu_i, \nu_j) = 1 - \frac{|K(d|a = \nu_i) - K(d|a = \nu_j)|}{Ka_{\max} - Ka_{\min}} \quad \text{(see [HuCe95])}, \qquad (n(a) = 6)$$

where $K(d|a = \nu_i) = \sum_{k=1}^{r(d)} P(d = k|a = \nu_i) \log_2 \frac{P(d=k|a=\nu_i)}{P(d=k)}$ and $Ka_{\max}$, $Ka_{\min}$ are the maximum and minimum K values as defined in formula of the attribute a.

To compute similarity between two objects we use the following coefficients:
$t \in [0, 1]$ – similarity threshold for objects,
$t(a) \in [0, 1]$ – similarity threshold for values of attribute a.
We assume that two values ν_i, ν_j of attribute a are similar when $s_a(\nu_i, \nu_j) \geq t(a)$. The Hamming distance between two objects $H_A(x, y)$ is the number of attributes from A where two objects have no similar values, i.e.
$H_A(x, y) = card(\{a \in A : \; s_a(a(x), a(y)) < t(a)\})$. We define

$$s_A^t(x, y) = \begin{cases} 1 & \text{if } 1 - \frac{H_A(x,y)}{card(A)} \geq t \\ 0 & \text{otherwise} \end{cases}.$$

We say that x and y are discernible when $s_A^t(x, y) = 0$.
Choosing an appropriate similarity measure can be done by performing experiments with a given decision table.

2 Rough set concept based on tolerance relations

In this section we present basic notions of the rough set concept based on tolerance relations. If $A = (U, A \cup \{d\})$ is a decision table and $B \subseteq A$ then $INF(B) = \{Inf_B(x) : x \in U\}$ is the set of information vectors $Inf_B(x) = \{(a, a(x)) : a \in B\}$. If $u \in INF(C)$ and $B \subseteq C \subseteq A$ then $u|_B = \{(a, w) \in u : a \in B\}$ i.e. $u|_B$ is the restriction of u to B. A tolerance decision table is defined by $(\boldsymbol{A}, \tau_A)$, where τ_A is a tolerance relation in $INF(A)$. We define tolerance set determined by an object x as follows:

$$TS(x) = \{y \in U : Inf_A(x)\tau_A Inf_A(y)\}.$$

If $t = 1.0$ and $t(a) = 1.0$ for all $a \in A$ then τ_A is an equivalence relation and it will be denoted by τ_1. Tolerance generalized decision is defined as follows:

$$\partial_{\tau_A}(x) = \{i : \exists x' \in U \ Inf_A(x)\tau_a Inf_A(x') \text{ and } d(x') = i\}.$$

Relative absorbent set
A subset $Y \subseteq U$ is a relative absorbent set for (τ_A, d) where τ_A is a tolerance relation iff
1) for each $x \in U$ there exists $y \in Y$ such that $Inf_A(x)\tau_A Inf_A(y)$ and $d(x) = d(y)$,
2) for every proper subset $Y' \subset Y$ condition 1) is not true.
In the paper we are interested in minimal (with respect to cardinality) relative absorbent sets.

The set $POS(\tau_A, \{d\}) = \bigcup_{x \in U} \{TS(x) : \exists i TS(x) \subseteq Y_i\}$ is called τ_A - *positive region of* $\{Y_i : i = 1, \ldots, r(d)\}$. The coefficient $\gamma(\tau_A, \{d\}) = \frac{card(POS(\tau_A, \{d\}))}{card(U)}$ is called the quality of approximation of classification $\{Y_i : i = 1, \ldots r(d)\}$. It expresses the ratio of all τ_A - *correctly* classified objects to all objects in the table.

Proposition 1.1 For every tolerance relation τ_A the following conditions are true

a) $POS(\tau_A, \{d\}) = \bigcup_{x \in U} \{TS(x) : card(\partial_{\tau_A})) = 1\}$.
b) If for all $x \in U$ $card(\partial_{\tau_A}(x)) = 1$ then $\gamma(\tau_A, \{d\}) = 1$.
c) $0 \leq \gamma(\tau_A, \{d\}) \leq \gamma(\tau_1, \{d\}) \leq 1$.
d) It is possible that $Inf_A(x)\tau_A Inf_A(y)$ and $\partial_{\tau_A}(x) \neq \partial_{\tau_A}(y)$.

3 Searching for tolerance thresholds

In this section we present problem of finding the Optimal Tolerance Thresholds (OTT) and its effective solution based on genetic algorithms. The problem is formulated as follows:
Input:
1) decision table $\boldsymbol{A} = (U, A \cup \{d\})$
2) similarity measures $s_a : V_a \times V_a \rightarrow [0, 1]$ for all $a \in U$.

Output. A set $\{t\} \cup \{t(a) : a \in A\}$ of optimal thresholds. By optimal one can understand solution which satisfies different conditions. Actually we would like to obtain maximization of the following function:

$$\frac{card(\tau_A \cap \{(x,y) : d(x) = d(y)\})}{card(\{(x,y) : d(x) = d(y)\})} + \gamma(\tau_A, \{d\}).$$

Before presenting some solution we introduce some notions:

Connections. We use the notion of connections to express the indiscernibility of objects. We inherit it from very simple observation, that if $x \in TS(y)$ then $y \in TS(x)$ and then we can say that there is a connection between x and y. We propose to discern two kinds of connection between objects: "good" and "bad".

∘ x and y have good connection $\Leftrightarrow x \in TS(y)$ and $d(x) = d(y)$

∘ x and y have bad connection $\Leftrightarrow x \in TS(y)$ and $d(x) \neq d(y)$

Thresholds matrices. The thresholds matrix $TM(a)$ for $a \in A$ is $card(U) \times card(U)$ matrix, where $TM_{ij}(a) = s_a(a(x_i), a(x_j))$ - the highest value of threshold, which cause that x_i, x_j are indiscernible. Such matrix is symmetric ($TM_{ij}(a) = TM_{ji}(a)$) and what is more important number of different values in $TM(a)$ is less or equal $1/2(k^2 - k) + 1$, where $k = card(V_a)$.

Tolerance sets matrices. Lets count tolerance sets separately for each $a \in A$. If we want to know $TS_a(x_i)$ we look for the i-th row in $TM(a)$ and if $TM_{ij}(a) \geq t(a)$ then $x_j \in TS_a(xi)$. We can build $n \times n$ matrix for saving TS_a in such way:

$$TSM_{ij}(a) = \begin{cases} 1 & \text{if } x_j \in TS_a(x_i) \\ 0 & \text{if } x_j \notin TS_a(x_i) \end{cases}.$$

If we increase the value of $t(a)$ then the TS_a will not change or become larger. So starting from $t(a) = 1.0$ and decreasing the value of threshold we can using above property find all values when TS_a changes. We can create lists of such thresholds for each a.

Lets assume that we have chosen all $t(a)$ for $a \in A$ and using this thresholds we have built all TS_a. Then we can find general tolerance sets by building $TSM(A)$:

$$TSM_{ij}(a) = \begin{cases} 1 & \text{if } \left(\sum_{a \in A} TSM_{ij}(a)\right) \geq t \cdot card(A), \\ 0 & \text{otherwise} \end{cases}$$

And of course if $TSM_{ij}(A) = 1$ it means that there is connection between x_i and x_j. This observation gives us a powerful tool to quick finding tolerance sets for given set of thresholds $\{t\} \cup \{t(a) : a \in A\}$.

It is easy to obtain all possible threshold value for attribute a from $TM(a)$. We think that we can throw out some values of threshold and don't consider them as interesting. Now we present a technique which help us to gain this aim.

1. We have descending list of all values from $TM(a)$ (1.0 is first). 2. For each value we can check what new connections appear when we decrease the threshold value from previous value in the list. It could be all good connections (G), all bad (B) or mixed (M) ones. So we can join with each value from list type of connections which this value

introduce. To decide, if value is "interesting" we use type of connection of it and its successor.
3. We start from the beginning of the list and use the key showed in the table below. In first column there are types of connections of value which is actually examined and in first row there are next types.

	G	M	B
G	Skip	Insert	Insert
M	Skip	Insert	Insert
B	Skip	Skip	Skip

The table should be read in such way: *if after G next is G then Skip value or if after G next is B then Insert value into new list*. To last values in the list which don't have successors we use rule as follows: if type≠B then Insert. As a result we build new list with some values from first one.

We repeat this process for each $a \in A$ and after that if we set the main threshold t we can check all possible combinations of thresholds to find out the best for our purpose. Of course it will be long process, because in the worst case for m attributes and only one t, the number of combinations is equal:

$$\frac{1}{2}\prod_{a \in A}\left(card(V_a)^2 - card(V_a)\right) + 1.$$

So it shows that we need some heuristics to find, maybe not the best of all, but very good solution in reasonable time. We think that genetic algorithm will be suitable for this purpose.

Genetic algorithm for OTT

We use standard schema of genetic algorithm (see [Mi94]).
Representation. The individuals are represented by number strings of length $card(A)$. Each position in chromosome corresponds indirectly with value of threshold for attribute (in the i-th position there is a number from threshold list for i-th attribute). For example:

attribute	list of values			number of values in the list
a_1	- 1.0 0.9 0.87 0.6 0.4			5
a_2	- 0.95 0.5			2
a_3	- 1.0 0.98 0.95 0.93 ... 0.32			35
chromosome -	2	2	4	
thresholds -	0.9	0.5	0.93	
	$t(a_1)$	$t(a_2)$	$t(a_3)$	

Initialization. For first population we used controlled random generator. It means that we accept only these individuals which have fitness greater or equal given threshold (for example ≥0,9). **Fitness function.** The fitness function depends on two parameters:

- the number of good connections between objects,
- the quality of approximation of classification

$$Fitness(\tau_A) = \frac{card(\tau_A \cap \{(x,y) : d(x) = d(y)\})}{card(\{(x,y) : d(x) = d(y)\})} + \gamma(\tau_A, \{d\}).$$

Selection. We use modified tournament selection algorithm. It means that to select one chromosome we randomly choose k individuals from population (with equal probabilities) and then with probability P_t one with the best fitness wins or with probability $1 - P_t$ we select any from k.
Mutation. Standard mutation affect with probability P_m of mutation on a single position of chromosome. Mutation of one position means replacement existing number by randomly chosen.
Crossing-over. We used classical, two-point crossover for chromosomes selected with the probability of P_c. For example:

$$\begin{array}{ccccccccc} 3 & 5 & | & 0 & 23 & 3 & | & 4 \\ & & & \Uparrow & \Downarrow & & & \\ 5 & 1 & | & 1 & 3 & 7 & | & 4 \end{array} \quad \Rightarrow \quad \begin{array}{cccccccc} 3 & 5 & | & 1 & 3 & 7 & | & 4 \\ & & & & & & & \\ 5 & 1 & | & 0 & 23 & 3 & | & 4 \end{array}$$

To find out the optimal thresholds we repeat genetic algorithm for a few possible values of t. Then we can choose the most suitable solution from them. All essential values of t we can find out from the set $\{1 - i \cdot \frac{1}{m-1} : i = 0, K, card(a) - 1\}$. Let us observe that very low values of t are not interesting.

4 Data reduction based on relative reducts and relative absorbents

In this section we present methods of attribute set and object set reduction. First we present problem of data reduction and introduce some preliminary definitions.
Input:
1) decision table $\boldsymbol{A} = (U, A \cup \{d\})$
2) similarity measures $s_a : V_a \times V_a \to [0,1]$ for all $a \in U$
3) a set $\{t\} \cup \{t(a) : a \in A\}$ of tolerance thresholds

Output: reduced decision table $\boldsymbol{A}' = (U', A' \cup \{d\})$ such that, $U' \subset U$ is a relative absorbent set of $(U, A \cup \{d\})$ and $A' \subset A$ is a relative reduct of $(U', A \cup \{d\})$.

We consider a special case of relative reducts which can be constructed from discernibility matrix DM by adding some constraints on the reduction of attributes from the entries of DM in such a way that in any entry after reduction there is enough attributes to discern between corresponding objects [KPSS95].

In practice the process of reduction of attributes set is organized as follows. First we modify the discernibility matrix. We set an entry to be empty in DM for any entry corresponding to two non discernible objects.

$$DM_{ij} = \begin{cases} \{a \in A : s_a(a(x_i), a(x_j)) < t(a)\} & \text{if} \quad s_A^t(x_i, x_j) = 0 \wedge d(x_i) \neq d(x_j) \\ \emptyset \quad \text{if} \quad s_A^t(x_i, x_j) = 1 \vee d(x_i) = d(x_j) & \end{cases}$$

Next we make reduction of superfluous entries in DM. We set an entry to be empty if it is a superset of another non-empty entry. At the end of this process we obtain the set $COMP$ of the so called components.

The described type of reducts can be generated by applying Boolean reasoning [SkSt94]. Here we present heuristics for computing one reduct of the considered type with the minimal number of attributes. These heuristics can produce sets which are supersets of considered reducts but the heuristics are more efficient than the general procedure.

First we introduce a notion of a minimal distinction. By *minimal distinction* (md, in short) we understand minimal set of attributes sufficient to discern between two objects. Let us observe that minimal component $comp$ consists of minimal distinctions and $card(comp)$ is equal or greater than $card(md)$[KPSS95]. We say that md is *indispensable* if there is a component composed out of only one md. We include all attributes from indispensable md to R.
Then from $COMP$ we eliminate all these components which have at least one md equal to md in R. It is important that the process of selecting attributes to R will be finished when the set $COMP$ will be empty. We calculate for any md from $COMP$:

$$c(md) = w_1 c_1(md) + w_2 c_2(md), \quad \text{where}$$

$$c_1(md) = \left(\frac{card(md \cap R)}{card(md)}\right)^p$$

$$c_2(md) = \left(\frac{card(comp \in COMP: \ \exists_{md' \subset comp} md' \subset (R \cup md))}{card(COMP)}\right)^q$$

for some natural numbers p and q, for example we can assume $p = q = 1$.
The first function is a "measure of extending" R. Because we want to minimize cardinality of R we are interested in finding md with the largest intersection with actual R. In this way we always add to R almost minimal number of new attributes.
The second measure is used to examine our profit after adding attributes from md to R. We want to include to R the most frequent md in $COMP$ and minimize $COMP$ as much as it is possible. When $c_2(md) = 1$ then after "adding this md" to R we will obtain "pseudo-reduct" i.e. it can be a superset of a reduct.

The heuristic for finding out one relative absorbent set is similar to the presented for computing one relative reduct [KPSS95]. The main difference is in the way in which we calculate and interpret components. In this case we do not build the discernibility matrix, but we replace it by a similar table containing for any object x_i all objects similar to x_i and with the same decision.

$$ST(x_i) = \{x_j : \ s_A^t(x_i, x_j) = 1 \text{ and } d(x_i) = d(x_j)\}$$

We make the same reduction as in discernibility matrix and we obtain components as essential entries in ST. For $COMP$ we can apply the algorithm used to compute a reduct assuming $card(md) = 1$. We add to the constructed relative absorbent set any object which is the most frequent in $COMP$ and then eliminate from $COMP$ all components having this object. This process terminates when $COMP$ is empty.

5 Tolerance decision rules

In this section we present generation of tolerance decision rules for consistent decision tables (i.e. $card(\partial_{\tau_A}(x)) = 1$ for all $x \in U$). For inconsistent decision table we construct decision rules with generalized decision.
A tolerance decision rule for a tolerance decision table $(A, \tau_A) = ((U, A \cup \{d\}), \tau_A)$ is any expression of the form $\alpha \Rightarrow (d, i)$, where $i \in V_d$ and α is a Boolean combination of tolerance descriptors i.e. expressions (a, v, r) where $a \in A$, $v \in V_d$ and $r \in [0,1]$. If α is a Boolean combination of descriptors then by $\alpha_{(A,\tau_A)}$ we denote the meaning of α in the tolerance decision table i.e. the set of objects in U with property α, defined inductively as follows:

(1) if α is of the form (a, v, r) then $\alpha_{A,\tau_A} = \{x \in U : s_a(a(x), v) \leq r\}$;
(2) $(\alpha \wedge \alpha')_{A,\tau_a)} = \alpha_{(A,\tau_A)} \cap \alpha'_{(A,\tau_A)}$; $(\alpha \vee \alpha')_{(A,\tau_A)} = \alpha_{(A,\tau_A)} \cup \alpha'_{(A,\tau_A)}$.

The decision rule $\alpha \Rightarrow (d, i)$ for (A, τ_A) is true in (A, τ_A) iff $\alpha_{(A,\tau_A)} \subseteq ((d, i))_{(A,\tau_A)}$; if $\alpha_{(A,\tau_A)} = ((d, i))_{(A,\tau_A)}$ then we say that rule is (A, τ_A) - *exact*.
Let $(A, \tau_A) = ((U, A \cup \{d\}), \tau_A)$ be tolerance decision table and its relative tolerance discernibility matrix is defined as in previous section. For each object x_i we build tolerance decision matrix TDM_i as follows:

$$TDM_i(j) = \{(a, a(x_i), t(a)) : a \in DM_{ij}\}.$$

Using those tables we construct tolerance decision functions in analogous way as the discernibility function is constructed from the discernibility matrix:

$$tdf(x_i) = \bigwedge\{\vee TDMi(j) : TDMi(j) \neq \emptyset\}.$$

Next we change form of tdf to disjunction of conjunctions and obtain prime implicants $\pi_{i1}, \ldots, \pi_{ik}$. We construct decision rules:

$$\pi_{i1} \Rightarrow (d, d(x_i)),$$
$$\ldots$$
$$\pi_{ik} \Rightarrow (d, d(x_i)).$$

In this way we can find out rules for all examples of particular decision and for all decisions. The rules learned from examples are then applied to support decisions concerning classification of new objects. This is done by matching the classified object's description by values of attributes to one of the rules.
We formulate problem of classifying new objects as follows:
Input:
1) the set of tolerance decision rules,

2) similarity measures $s_a : V_a \times V_a \to [0,1]$ for all $a \in U$,
3) new object x,
Output: decision $d(x)$.

Let a given new object x be described by values $\{a(x) : a \in B\}$ of the attributes B occurring in the condition part of the rule $\alpha \Rightarrow d = i$ described by descriptors $\{(a, v_a, t(a)) : a \in B\}$.
For such rule we count measure of matching

$$match(x, \alpha \Rightarrow d = i) = \frac{card(\{a \in B : \ s_a(a(x), v_a) \geq t(a)\})}{card(B)}$$

We use a few strategies for choosing decision $d(x)$:

- most frequent decision from the set of rules with the best *match*
- most frequent decision from the set of rules with *match* $> t_M$ for some natural number $t_M < card(A)$.

6 Examples

For the purpose of demonstration of presented approach we use example.
The decision table $BANK95 = (U, A \cup \{d\})$ presented in Table 1 profiles twenty hypothetical bank customers. Attribute Decision means evaluation of a past credit record of a customer in terms of three qualitative categories: very good (3), good (2), poor (1). Similarity measures are included at the top of the Table 1.

First we search for tolerance thresholds. We use genetic algorithm described previously. We use such parameters: 60 chromosomes in population, 100 populations, $P_m = 0.05$, $P_t = 0.9$, $P_c = 0.8$, $k = 8$ chromosomes in each tournament. The best chromosome has the value of fitness function equal 1.13. We obtain thresholds presented below.

t	0.86						
t(a)	1.0	0.5	0.5	0.86	1.0	0.85	0.75

Now we can count $ST(x)$ for all objects x:

ST(1)	1 2 4 5	ST(6)	2 4 5 6	ST(11)	10 11	ST(16)	16
ST(2)	1 2 5 6	ST(7)	7	ST(12)	12	ST(17)	13 17
ST(3)	3 4 8	ST(8)	3 8	ST(13)	13 17	ST(18)	15 18 19
ST(4)	1 3 4 6	ST(9)	9	ST(14)	14	ST(19)	15 18 19
ST(5)	1 2 5 6	ST(10)	10 11	ST(15)	15 18 19	ST(20)	20

Next we start process of data reduction. We use the heuristics based on adding to the constructed absorbent, objects which are the most frequent in components not used so far. In this way we are adding in any iteration step one new object to the constructed absorbent and we eliminate all components including this object. We obtain the set of components: $COMP = \{(2\ 4\ 5\ 6), (1\ 3\ 4\ 6), (1\ 2\ 5\ 6), (1\ 2\ 4\ 5), (15\ 18\ 19), (13\ 17), (10\ 11), (3\ 8), (20), (16), (14), (12), (9), (7)\}$.

n(a)	1	5	4	1	2	1	1	1
Number of object	**Year of birth**	**City**	**Type of employment**	**Salary**	**Children**	**Account**	**Credit**	**Decision**
1	1960	Warsaw	T	3000	2	2000	15000	1
2	1960	Cracow	T	2300	1	500	8000	1
3	1955	Cracow	T	4000	1	300	6000	1
4	1950	Bialystok	T	3200	3	2000	10000	1
5	1960	Warsaw	P	2200	2	2500	3500	1
6	1966	Bialystok	T	2750	3	800	4500	1
7	1949	Bialystok	P	6000	1	8000	8000	1
8	1965	Cracow	T	4300	4	1700	3300	1
9	1942	Warsaw	P	4500	6	5000	3000	2
10	1955	Cracow	P	8000	3	7200	13000	2
11	1952	Bialystok	P	7200	3	6000	10000	2
12	1962	Cracow	T	2800	1	3000	5000	2
13	1948	Bialystok	P	8500	2	9500	15000	3
14	1947	Bialystok	T	7900	2	12000	25000	3
15	1960	Cracow	P	5000	0	7500	9000	3
16	1958	Warsaw	P	4800	0	4000	7000	3
17	1948	Bialystok	T	4000	2	9800	10500	3
18	1962	Bialystok	P	5000	0	7000	6000	3
19	1962	Warsaw	P	5450	0	6800	11000	3
20	1965	Warsaw	P	4600	2	13400	10000	3

Table 1.

Finally we choose the set of following objects: $\{1\ 2\ 3\ 7\ 9\ 10\ 12\ 13\ 14\ 15\ 16\ 20\}$.
Now we search for relative tolerance reduct. We obtain the set of components:
$COMP = \{(1\ 5), (1\ 4), (4\ 6), (1\ 6), (1\ 7)\}$.
All minimal distinctions in components are indispensable, hence we obtain one reduct $\{1\ 4\ 5\ 6\ 7\}$.
Next we compute decision rules for reduced decision table. We present only examples:
$(Account,\ 500,\ 0.85) \Rightarrow d = 1$
$(Salary,\ 3000,\ 0.86) \wedge (Credit,\ 15000,\ 0.75) \Rightarrow d = 1$
$(Salary,\ 8000,\ 0.86) \wedge (Account,\ 7200,\ 0.85) \Rightarrow d = 2$
$(Year,\ 1955,\ 1.0) \wedge (Salary,\ 8000,\ 0.86) \Rightarrow d = 2$
$(Account,\ 12000,\ 0.85) \Rightarrow d = 3$
$(Salary,\ 4800,\ 0.86) \wedge (Children,\ 0,\ 1.0) \wedge (Account,\ 4000,\ 0.85) \Rightarrow d = 3$ □

Conclusions

This paper has focused attention on data reduction methods and decision rules' generation. We proposed a new technique which exploits rough set theory based on tolerance relations and Boolean reasoning. The decision rules are generated from the reduced data table by applying Boolean reasoning methods developed in [Sk93]. The object reduction eliminates objects which are very close (with respect to the tolerance distance) to the remaining objects in the absorbent set. Attribute reduction is done without changing the classification quality. Hence one can expect that the classification quality (of the unseen so far objects) obtained by tolerance decision rules generated from the reduced decision table is very close to the quality of classification of the rules generated from the original decision table. We are currently testing the presented algorithms on some real databases and hope to report our results in the future.

References

[Br90] Brown F.N., Boolean Reasoning. Kluwer, Dordrecht 1990.

[HuCe95] Hu X., Cercone N., Rough Sets Similarity-Based Learning from Databases, Proceedings of the First International Conference on Knowledge Discovery and Data Mining, Montreal, Canada, August 20- 21 1995.

[KPSS95] Krętowski M., Polkowski L., Skowron A., Stepaniuk J., Data Reduction Based on Rough Set Theory, Proceedings of the International Workshop on Statistics, Machine Learning and Knowledge Discovery in Databases, Crete, Greece, April 28-29 1995.

[Mi94] Michalewicz Z., Genetic Algorithms + Data Structures=Evolution Programs, Springer Verlag 1994.

[Pa91] Pawlak Z., Rough Sets. Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, Dordrecht, 1991.

[ShLa90] Shrager J., Langley P., Computational Models of Scientific Discovery and Theory Formation, Morgan Kaufmann, San Mateo CA 1990.

[Sk93] Skowron A., A Synthesis of Decision Rules: Applications of Discernibility Matrices, Proc. of the Conference on Intelligent Information Systems, Augustow, Poland, June 7-11, 1993.

[SkSt94] Skowron A., Stepaniuk J., Generalized Approximation Spaces, ed. T.Y.Lin, Proceedings of the Third International Workshop on Rough Sets and Soft Computing, San Jose, California, USA, November 10-12, 1994, 156-163.

[Sl92] Słowiński R., (ed.) Intelligent Decision Support - Handbook of Applications and Advances of the Rough Sets Theory. Dordrecht: Kluwer 1992.

[Te94] Tentush I., On Minimal Absorbent Sets for some types of Tolerance Relations. Research Report ICS 23/94. Warsaw University of Technology.

[Zi94] Ziarko W., (ed.) Rough Sets, Fuzzy Sets and Knowledge Discovery, Springer Verlag 1994.

[Zy92] Zytkow J., Proceedings of the ML-92 Workshop on Machine Discovery.