Knowledge acquisition method for relational knowledge representation

Jakub M. Tomczak ∗ Jerzy Świątek ∗

* Institute of Informatics, Wrocław University of Technology, Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland (e-mail: jakub.tomczak@pwr.wroc.pl).

Abstract: In this paper a new method for knowledge extraction was presented. First, the relational knowledge representation with all crucial definitions was introduced. Second, problem statement for knowledge acquisition using relational knowledge representation was formulated. Then the modified coverage strategy was proposed. At the end a benchmark problem was considered and results were presented.

Keywords: Knowledge acquisition, Knowledge representation, Relational description.

1. INTRODUCTION

Nowadays in each minute many data are collected by institutions like hospitals, research laboratories, shops, banks, stores, and so one (Frawley et al. (1992)). In each situation, beside handling and transferring data, there is a need of extracting useful information, hidden relationships or knowledge in data, e.g. for creating a credit profile of clients’ groups in a bank, looking for not obvious patterns in chemical or biological substances, classification of illnesses by using different symptoms, seeking unnatural patterns in network’s activity (Pal and Jain (2005)). And because of huge amount of data this need cannot be fulfilled even by experts who are not able to analyze those data in acceptable time and with acceptable quality (so called knowledge acquisition bottleneck, Lavrac and Dzeroski (1994); Michalski (1984)).

Therefore, there is a big demand for efficient analysis and understanding for huge amount of data. Knowledge acquisition (KA) and data mining (DM) methods meet this claim. Moreover, because of this need, KA became one of the most important issues in modern computer science (Cichosz (2000); Cios et al. (2007); Frawley et al. (1992); Larose (2005); Shi (1992); Witten and Frank (2005)). There are used different method to obtain knowledge: rules-based (e.g. AQ-family methods, Michalski (1984)), tree-based (e.g. ID-family methods, Shi (1992)), inductive logic programming (Lavrac and Dzeroski (1994)), knowledge discovery (e.g. BACON, COPER, Shi (1992)), neural networks (e.g. backpropagation method, Witten and Frank (2005)), and many others. However, there is still a need to improve accuracy of KA methods.

Therefore, in this paper we consider a new method for relational knowledge representation (Bubnicki (1993; 2004); Bubnicki and Szlachetko (1993); Świątek (1989); Tomczak (2008); Tomczak and Świątek (2008)). In general the relational knowledge representation (RKR) gives opportunity to formulate a logical description (Bubnicki and Szlachetko (1993); Świątek (1989)) and non-functional numerical descriptions, also known as relational patterns (Bubnicki (1993); Frawley et al. (1992); Świątek (1989); Tomczak and Świątek (2008), about considered system. The novelty of that approach is to include non-functional description in the logical expression. It gives possibility to separate examples not only by simple inequality as it is made now (e.g. AQ methods, ID methods) but by using more sophisticated expressions (e.g. inequality of ellipse, net of perceptrons). However, for system described by relational knowledge representation there is a need to create dedicated algorithms and methods for different system’s models.

This paper is constructed as follows. In Section 2 crucial definitions and relational knowledge representation are introduced. Next, in Section 3, general problem for knowledge acquisition is stated. In Section 4 a new method for knowledge acquisition using relational knowledge representation is proposed. Later, in Section 5, a benchmark dataset is considered and our method was compared with well-known data mining algorithms. At the end, in Section 6, some conclusions are drawn.

2. DEFINITIONS

2.1 Knowledge acquisition process

In the literature different conceptions about knowledge are used (see Bubnicki (2004); Cios et al. (2007); Mitchell (1997); Witten and Frank (2005)). However, in the field of knowledge engineering (Jagielski (2001)), knowledge is defined as the result of the following process:

Data Processing Information Evaluation Knowledge

It means that knowledge is conclusions from data and information or information with some context. Moreover, in the field of machine learning it is said that knowledge has to be understandable for a machine and human being as well (Lavrac and Dzeroski (1994); Michalski (1984); Mitchell (1997)). Therefore, there is a need to stress the representation that is used. Different knowledge representation has different expression possibilities.
Knowledge is the final product of knowledge acquisition process. However, it is still quite ambiguous what knowledge means exactly. In inductive learning (e.g. Cios et al. (2007); Cichosz (2000); Lavrac and Dzeroski (1994); Mitchell (1997); Shi (1992); Witten and Frank (2005)) it is assumed that there exists concept which has to be learned from data. Concept is chosen physical quantity about some system which could be measured or observed. In more formal way (e.g. Maloof (2003)), let us consider a system with input \(x \in X\) and output \(y \in Y\) which is a finite set of values (in classification problems it is called class). Then concept is a mapping \(f : X \rightarrow Y\).

However, in reality we usually do not know the concept and we are interested in finding some approximation of this concept, \(\hat{f}_a\). This approximation is formalized in some representation, \(R\), so called knowledge representation. There are many of such formalizations, for example: artificial neural networks (e.g. Shi (1992)), classification rules (e.g. Witten and Frank (2005)), trees (e.g. Cichosz (2000)), probability distributions (e.g. Bubnicki (2004)), logical expressions (e.g. Lavrac and Dzeroski (1994)), and others (frames, ontologies, e.g. Jagielski (2001)).

The best concept approximation is found by using inference rules (KA method), \(\Psi\). Inference rules use observations and due to them look for best parameters (e.g. parameters’ values and/or structure of the representation) of the approximation so that some evaluation criteria are fulfilled or/and have highest/lowest values.

### 2.3 Background knowledge

In many problems of reasoning is is assumed that there is already some knowledge about the problem. Such knowledge is called background knowledge which is a set of given a priori knowledge connected with the the considered concept and could be represented in following ways (Lavrac and Dzeroski (1994); Mitchell (1997); Shi (1992)):

- **extensional** – a list of facts about the concept;
- **intensional** – a list of definitions and/or descriptions of a language describing the concept (e.g. description of a rule by other rules);
- constraints;
- other (e.g. which attributes should be treated as an input, and which as an output).

### 2.4 Concept learning – problem statement

In the problem of concept learning we are interested in finding the best approximation in chosen knowledge representation, so that it gives the best generalization to the observation. It could be stated as follows (e.g. Shi (1992)):

**Given:**

- Observations (training examples, test examples).
- Knowledge representation \(R\).
- Background knowledge.
- Inference rules (knowledge acquisition method).
- Evaluation criteria (assessment of concept approximation).

**Find:**

- The best, due to evaluation criteria, concept approximation \(\hat{f}_a\) (in other words – the best parameters of chosen knowledge representation).

### 2.5 Relational system

By the relational system we understand a complex concept about some physical quantity in which an input and an output can be distinguished and which are connected by some relation. In other words - input and output of the system are connected by some relation and the system is represented by relational description.

In formal way we define relational system as follows. Let us consider a static system in which an input (input attributes), \(x \in X\), which is a vector \(x = (x^{(1)}, x^{(2)}, \ldots, x^{(S)})\) (i.e. \(X\) could be finite or continuous \(X \subseteq \mathbb{R}^S\)), and an output (output attributes), \(y \in Y\) (we assume that \(Y = (y_1, y_2, \ldots, y_L)\) but it could be also \(Y \subseteq \mathbb{R}^L\) and then output is also a vector), can be distinguished. Then the relational system is described by a set of properties (predicates) concerning \((x, y)\): \(\mathcal{R}(x, y)\) denotes the set of all pairs \((x, y)\) for which properties are satisfied (Bubnicki (1993, 2004); Rasiowa (1971)),

\[
\mathcal{R}(x, y) = \{(x, y) \in X \times Y : w[\Phi(x, y)] = 1\}
\]

where:

- \(w[\cdot]\) – a logic value, \(w[\cdot] \in \{0, 1\}\);
- \(\Phi(x, y)\) – a complex predicate describing the object that consists of \(M\) single predicates, \(\{\phi_1(x, y), \ldots, \phi_M(x, y)\}\), for which following expression holds true:

\[
w[\Phi(x, y)] = w[\bigwedge_{m=1}^M \phi_m(x, y)].
\]

and

\[
w[\bigwedge_{m=1}^M \phi_m(x, y)] = 1.
\]

**Remark.** The single predicate, \(\phi_m\), is built of logic operators \(\land\) and \(\lor\), and \(-\), and can concern the input property (e.g. \(\phi_x = \"x > 20\"\)), the output property (e.g. \(\phi_y = \"y = 1\"\)), or input-output property (e.g. \(\phi = \"y > 2x + 1\"\)).

In many cases only a model of the system is known, \(\mathcal{R}_a(x, y) \subseteq \mathcal{R}(x, y; a)\). In other words, an expert or a reliable source gives the description of the system known in accuracy to unknown parameters, \(a \in A\) (parameters could be expressed in numerical way \(A \subseteq \mathbb{R}^K\), e.g. values of an algebraic expression, or/and could be connected with some structure, e.g. number of conditions in the rule). This description is a set of true logical statements (predicates) about \(x\) and \(y\), \(F(x, y; a)\).

Thus the definition of the relational model is following

\[
\mathcal{R}_a(x, y; a) = \{(x, y) \in X \times Y : w[F(x, y; a)] = 1\}
\]

where:

- \(a\) – unknown parameters, \(a \in A\);
- \(x\) – input;
- \(y\) – output;
- \(F(x, y; a)\) – description of system’s properties.
Example 1. The model could be an inequality of the ellipse (non-functional numerical relation):
\[ \mathcal{R}_a = \{ (x, y) : w[F(x, y; a) \equiv \left( \frac{x^2}{a_1^2} + \frac{y^2}{a_2^2} \leq 1 \right)] = 1 \} \tag{3} \]
where \( a_1 \) and \( a_2 \) are unknown parameters.

Example 2. The model could be given in logical form, i.e. classification rule,
\[ \mathcal{R}_a = \{ (x, y) : w[F(x, y; a) \equiv (B(x) \Rightarrow C(y))] = 1 \} \tag{4} \]
where \( B(x) \) determines conditions (so called body) and \( C(y) \) - conclusions. Of course there are many other possible models (e.g. \( B(x) \Rightarrow C(x, y) \), \( B(x) \wedge C(Y) \Rightarrow H \), where \( H \in \{0, 1\} \).

2.6 Relational knowledge representation

There are two possible situations: 1) the relation is known, 2) only the relational model is known. In the first case the relational knowledge representation is defined as follows:
\[ RKR = \langle B, X, Y, \mathcal{R} \rangle \tag{5} \]
where:
- \( B \) - a background knowledge;
- \( X \) - an input space;
- \( Y \) - an output space;
- \( \mathcal{R} \) - a relation.

Otherwise there is:
\[ RKR = \langle B, X, Y, A, \mathcal{R}_a \rangle \tag{6} \]
where:
- \( A \) - a space of unknown parameters;
- \( \mathcal{R}_a \) - a model of the relation.

2.7 Dataset

Dataset consists of two collections of data (in mathematical sense – multisets):
- training set – used in learning stage;
- test set – used in evaluation stage.

In our considerations by training set we understand a sequence of observations, called also examples or instances,
\( (x_n, y_n, \lambda_n), n = 1, 2, \ldots, N, \) which represent measured or observed relation. We denote training set as follows:
\[ \mathcal{R}_N = \{ (x_n, y_n, \lambda_n) \in X \times Y \times \Lambda : n = 1, 2, \ldots, N \} \tag{7} \]
where:
- \( x_n \) - input values of \( n^{th} \) instance;
- \( y_n \) - output values of \( n^{th} \) instance;
- \( \lambda_n \) - belonging of \( n^{th} \) instance to the relation, \( \lambda_n \in \Lambda \equiv \{0, 1\} \).

Thus we can divide \( \mathcal{R}_N \) into two subsets:
\[ \mathcal{R}_N = \mathcal{R}_{N,0} \cup \mathcal{R}_{N,1} \]
where:
- \( \mathcal{R}_{N,0} = \{ (x_n, y_n, \lambda_n) \in \mathcal{R}_N : \lambda_n = 0 \} \) - so called negative examples;
- \( \mathcal{R}_{N,1} = \{ (x_n, y_n, \lambda_n) \in \mathcal{R}_N : \lambda_n = 1 \} \) - so called positive examples.

Obviously for those sets the expression \( \mathcal{R}_{N,0} \cap \mathcal{R}_{N,1} = \emptyset \) holds true.

Moreover, we assume that in the Data Processing stage of Knowledge Acquisition Process data are denoised and missing values are already handled. Furthermore, training set is more or less complete specification of the relation.

2.8 Measures for relational model

In the literature there are many measures which determine correctness of the knowledge representation (see e.g. Witten and Frank (2005)). However, for logical models (e.g. rules) especially two of them are frequently used – accuracy, \( \mu_a \), and coverage, \( \mu_c \). They are defined as follows:
\[ \mu_a(F, E_N) = \frac{\text{card}(E_B \cap E_{N,1} \cap E_{B})}{\text{card}(E_B)} \tag{8} \]
\[ \mu_c(F, E_N) = \frac{\text{card}(E_B \cap E_{N,1} \cap E_{B})}{\text{card}(E_{N,1})} \tag{9} \]
where \( \text{card}() \) – cardinality of the set,
\( E_N \) - a set of indexes of all examples,
\( E_{N,1} \) - a set of indexes of positive examples,
\( E_B \) - a set of indexes of examples that are not in contradiction to the background knowledge,
\( E_{B} \) - a set of indexes of examples covered by proposed description.

Remark. In the inductive concept learning very important measure is also the degree of generalization. But this measure is not considered in this paper.

3. KNOWLEDGE ACQUISITION PROBLEM STATEMENT

The general problem statement for knowledge acquisition in relational system is following:

Given:
- Training set, \( \mathcal{R}_N \).
- The background (a priori) knowledge.
- The relational knowledge representation with model (model is given a priori for the considered problem), \( RKR \).

Find:
- Unknown parameters in description \( F(x, y; a) \) which covers as much positive examples as possible with highest accuracy.

There are two possible situations we are considering:
1) If non-functional numerical model is known, then numerical parameters have to be found.
2) If logical model which consists non-functional numerical model is known, then the predicates which represents each class have to be found.

In the first problem a parameter estimation or an optimal model selection methods for relational knowledge representation could be applied (Świątek (1989); Tomczak (2008); Tomczak and Świątek (2008)). These methods are different than these for traditional functional description because the idea is the same – a model or class of models is given and parameters' values have to be found which fit the best to observations (due to some criterion).
In the second case some new method has to be proposed because so far no such algorithms were formulated. However, if we deal only with logical models without consisting non-functional models then AQ-family methods or ID-Tree-based methods could be applied (see e.g. Cichosz (2000); Cios et al. (2007); Shi (1992)).

4. KNOWLEDGE ACQUISITION METHOD

Let us introduce some method for knowledge acquisition for logical model in form of implication. Such method has to formulate proposed description using implications that body consists input and conclusion - output. Moreover, numerical input attributes can be treated as partial numerical relations (however, the numerical model has to be given first). Furthermore, such proposed description has to have highest values of accuracy and coverage.

To fulfill all requirements we propose modified coverage strategy. Coverage strategy is a well-known method for formulating rules (see e.g. Witten and Frank (2005)).

In our approach additional step connected with partial numerical relations is included. Moreover, we use both measures - accuracy and coverage, not only coverage as it takes place in the basic coverage method.

Remark. Because of lack of space, descriptions of Parameter Estimation and Optimal Model Selection cannot be presented in this paper, but they can be found e.g. in Bubnicki (1993); Bubnicki and Szlachetko (1993); Swiatek (1989); Tomczak (2008); Tomczak and Swiatek (2008).

Remark. Simple formula is an attribute which equals to some value (Bubnicki (2004)). Simple formula is denoted by $\alpha$. In the literature it is also known as atomic formula of attributive logic (Michalski (1984)).

Remark. For the distinction, numerical relation in the procedure is called partial relation, $\tilde{R}$, and $\tilde{F}$.

4.1 Procedure

INPUT: $\mathcal{R}_N$ - dataset, $X_{\text{num}}$ - numerical part of input, $X_{\text{nom}}$ - nominal part of input ($X = X_{\text{num}} \times X_{\text{nom}}$), output $y \in \{y_1, y_2, \ldots, y_L\}$, model for numerical part - $\mathcal{R}_a(\cdot, \cdot)$, logical model in form of implications like (4), $Q(\cdot)$ - evaluation criterion (if needed).

START: $F(x, y; a) \equiv \emptyset$.

STEP 1

If $X_{\text{num}} \neq \emptyset$, then take the numeric part of input, Set part of them as "input", $x_{\text{num}, \text{in}} \equiv \bar{x}$, and rest as "output", $x_{\text{num, out}} \equiv \bar{y}$.

Else $k = 1$ and GOTO STEP 3.

STEP 2

If $K = 1$, then apply Parameter Estimation, $\Psi_1$. Result is $\Psi_1(\mathcal{R}_N, A) = a_N$ and partial relation $\mathcal{R}_{a_N}(\bar{x}, \bar{y})$.

Else:

For each class $y_l$, $l = 1, 2, \ldots, L$, apply Optimal Model Selection, $\Psi_2$, with $\lambda_n = 1$ if class of $n^{th}$ instance $y_n = y_l$ and $\lambda_n = 0$ - otherwise. Result is $\Psi_2(\mathcal{R}_N, A) = a_N$ and partial relation $\mathcal{R}_{a_N}(\bar{x}, \bar{y})$.

Set $k = 1$.

STEP 3

If $l > L$, then GOTO STEP 6.

Else create a partial relation description $\tilde{F}_l(x, y; a_N)$ in form:

$$(B_l(x) \equiv \emptyset) \Rightarrow (C(y) = y_l).$$

STEP 4

For class $y_l$ calculate $\mu_c$ and $\mu_a$ for all nominal attributes with different values and for distinct partial relation,

$$\tilde{R}^{(l)}_{a_N} = \mathcal{R}^{(l)}_{a_N} \setminus \bigcup_{p \neq l} \mathcal{R}^{(p)}_{a_N}.$$ 

Add to the body this attribute value or partial relation which has the highest $\mu_c$.

To the chosen attribute value or partial relation add as a disjunction all simple formula for which $\mu_a = 1$.

Calculate $\mu_c$ and $\mu_a$ of $\tilde{F}_l$.

STEP 5

If there is no simple formula left, then $\tilde{F}_l(x, y; a_N) = "B_l(x) \Rightarrow y_l", l = l + 1$ and GOTO STEP 3.

Else take not yet added simple formula, $\alpha$, with highest $\mu_c$ and add it to the body of description as a conjunction, $B_l(x) \land \alpha$. Calculate $\mu_c$ and $\mu_a$.

If $\mu_c$ and $\mu_a$ have risen or are the same as previous, then GOTO STEP 5.

Else remove formula $\alpha$ and GOTO STEP 5.

STEP 6

Bound all implications as a conjunction, $l = 1, 2, \ldots, L$, and create final description:

$$F(x, y; a_N) = \tilde{F}_1 \land \tilde{F}_2 \land \cdots \land \tilde{F}_L \land \mathcal{B}$$

and STOP.

OUTPUT: $F(x, y; a_N)$

5. EXAMPLE

5.1 Dataset description and results

Our method was checked on the benchmark dataset called Weather (see e.g. Shi (1992); Witten and Frank (2005)). This dataset is presented in Table 1.

<table>
<thead>
<tr>
<th>No.</th>
<th>outlook</th>
<th>temperature</th>
<th>humidity</th>
<th>windy</th>
<th>play (class)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>sunny</td>
<td>80</td>
<td>85</td>
<td>FALSE</td>
<td>no</td>
</tr>
<tr>
<td>2.</td>
<td>sunny</td>
<td>80</td>
<td>90</td>
<td>TRUE</td>
<td>no</td>
</tr>
<tr>
<td>3.</td>
<td>rainy</td>
<td>65</td>
<td>70</td>
<td>TRUE</td>
<td>no</td>
</tr>
<tr>
<td>4.</td>
<td>overcast</td>
<td>83</td>
<td>86</td>
<td>FALSE</td>
<td>yes</td>
</tr>
<tr>
<td>5.</td>
<td>rainy</td>
<td>68</td>
<td>80</td>
<td>FALSE</td>
<td>yes</td>
</tr>
<tr>
<td>6.</td>
<td>sunny</td>
<td>69</td>
<td>70</td>
<td>FALSE</td>
<td>yes</td>
</tr>
<tr>
<td>7.</td>
<td>rainy</td>
<td>75</td>
<td>80</td>
<td>FALSE</td>
<td>yes</td>
</tr>
<tr>
<td>8.</td>
<td>sunny</td>
<td>75</td>
<td>70</td>
<td>TRUE</td>
<td>yes</td>
</tr>
<tr>
<td>9.</td>
<td>overcast</td>
<td>72</td>
<td>90</td>
<td>TRUE</td>
<td>yes</td>
</tr>
<tr>
<td>10.</td>
<td>overcast</td>
<td>81</td>
<td>75</td>
<td>FALSE</td>
<td>yes</td>
</tr>
<tr>
<td>11.</td>
<td>sunny</td>
<td>72</td>
<td>95</td>
<td>FALSE</td>
<td>no</td>
</tr>
<tr>
<td>12.</td>
<td>rainy</td>
<td>71</td>
<td>91</td>
<td>TRUE</td>
<td>no</td>
</tr>
<tr>
<td>13.</td>
<td>overcast</td>
<td>64</td>
<td>65</td>
<td>TRUE</td>
<td>yes</td>
</tr>
<tr>
<td>14.</td>
<td>rainy</td>
<td>70</td>
<td>96</td>
<td>FALSE</td>
<td>yes</td>
</tr>
</tbody>
</table>

Our aim is to determine when to play ($play = YES$) and when - not. Because the concept is described by numerical
and nominal attributes as well, so logical description with partial numerical description could be quite good concept approximation. After analysing the data, an inequality of the ellipse was chosen as a class of models for temperature, $x^{(2)}$, and humidity, $x^{(3)}$.

An example application of the KA method presented in Section 4.1 with Optimal Model Selection for inequality of the ellipse (for the description of the algorithm see Tomczak and Swiatek (2008)) for class play = YES is considered with examples 1-10 as training set and rest as test set (see Table 1). Then, in Step 2, we find an ellipse with parameters $a: \text{semimajor} = 12.1432$, $\text{semiminor} = 10.6226$, center = (73, 79.1), angle of rotation = 34.6.

In Step 3 we formulate logical expression for class play = YES. In Step 4 we calculate measures for ellipse: $\mu_c = \frac{7}{7}$, $\mu_a = \frac{7}{7}$, and for other attributes, outlook: overcast – $\mu = \frac{1}{7}$, sunny – $\mu = \frac{2}{7}$, rainy – $\mu = \frac{2}{7}$, windy: TRUE – $\mu = \frac{2}{7}$, FALSE – $\mu = \frac{5}{7}$.

Next we choose inequality of ellipse because of its measures value and we add to it, as a disjunction, overcast (it has $\mu = 1$). In Step 5 we add other attributes values, but $\mu_c$ and $\mu_a$ take smaller values.

Thus the final relation is following:
\[
\mathcal{R}_a(x, y) = \{(x, y) \in X \times Y : \tilde{F}_0 \land \tilde{F}_1 \land B\}
\]

where $B = \{(x, y) \in X \times Y : x^{(2)} \in [0, 100] \land x^{(3)} \in [0, 100]\}$;

$\tilde{F}_1 : \tilde{F}_a^{(1)}(x^{(2)}, x^{(3)}) \lor (\text{outlook} = "\text{overcast}") \Rightarrow \text{YES}$;

$\tilde{F}_0 : \tilde{F}_a^{(0)}(x^{(2)}, x^{(3)}) \Rightarrow \text{NO}$;

and $\tilde{R}_a^{(1)}(x^{(2)}, x^{(3)}) = \{(x, y) : \tilde{F}_a^{(1)}(x^{(2)}, x^{(3)})\}$ – the inequality of the ellipse, $\tilde{R}_a^{(0)} = X \setminus \tilde{R}_a^{(1)}$.

The results were compared with well-known data mining methods (see Cichosz (2000); Witten and Frank (2005)):

- JRip – RIPPER algorithm for fast, effective rule induction;
- NNge – Nearest-neighbor method of generating rules using nonnested generalized exemplars;
- OneR – one attribute classifier;
- PART – Obtain rules from partial decision trees built using J4.8;
- Ridor – Ripple-down rule learner;
- Naive Bayes;
- Multilayer Perceptron;
- J48 tree – C4.5 decision tree learner (implements C4.5 revision 8);

in the classification task. Performance measures were following: classification accuracy, true positive and false positive rates, and kappa statistics (for details see Witten and Frank (2005)). All methods, including ours, were implemented in WEKA software (see WEKA (2005)). Cross-validation with 10 folds was applied. All results are shown on Figures 1, 2, 3, 4.

5.2 Discussion

All performance measures show that our method gives the best results (as well as J48 algorithm). It has highest classification accuracy (see Figure 1) and highest true
positive rate (around 93%, see Figure 2) and false positive rate (100%, see Figure 3). Therefore, due to the inductive reasoning paradigm we can say that almost ideal description was found.

Moreover, the kappa statistics’ value is highest for relational description among other methods and equal with J48 algorithm (see Figure 4). Furthermore, kappa value equals 0.85 which implies that relational approach is a very good description for considered problem.

However, obtained results give only premise that our approach has good outcomes. That is why further investigations are needed. First, more theoretical analysis about the algorithm (e.g. algorithm’s complexity, working time) should be made. Second, more experiments, especially on real-life dataset, have to be carried out.

6. CONCLUSIONS AND FINAL REMARKS

In this paper the new method for knowledge extraction was presented. First the relational knowledge representation with all crucial definitions was introduced. Then problem statement for knowledge acquisition using relational knowledge representation was formulated. Later the modified coverage strategy was proposed. At the end the benchmark problem was considered and results were presented.

Obtained results showed that usage of relational knowledge representation and KA methods for RKR is very interesting and promising. Nevertheless, more experiments and research are needed. Furthermore, presented approach could be applied only for static systems. Therefore, dynamical systems have to be considered and incremental (on-line) learning system should be developed for streams of data (e.g. Maloof (2003)). Especially that in many real-life cases (e.g. anti-spam software, user behaviour learning) there is a need for constant decision-making on already extracted knowledge. Thus static approach could not be efficient due to enormous data accumulation and its transformation time.

Besides, there are several limitations of proposed approach:

(1) Class of models (e.g. inequality of the ellipse) should be chosen due to some data analysis.
(2) Procedures for parameter estimation or optimal model selection could be very complicated which extends working time of the whole KA method.
(3) Conditions about adding simple formulas could be too restricted.

However, some possible solutions could be proposed. The first problem could be solved by applying different class of models (e.g. inequality of ellipse, separating curve) and choosing this one with best criterion value. The last, third point, could be also easily solved by applying some additional algorithm’s parameters for threshold below which values of coverage and accuracy cannot get. Unfortunately, problem 2 cannot be avoided and e.g. application of Support Vector Machines would radically increase working time.

REFERENCES