CRITICAL ANALYSIS OF ROUGH SETS APPROACH TO MACHINE LEARNING

Igor Kononenko, Samo Zorc
University of Ljubljana, Faculty of electrical engineering & computer science,
Tržaška 25, SI-61001 Ljubljana, Slovenia
Phone: +386 61 1768390, Fax: +386 61 264990
e-mail: igor.kononenko@ninurta.fer.uni-lj.si

Keywords: rough sets theory, critical analysis, machine learning

Edited by: Rudi Murn

Received: March 9, 1994 **Revised:** June 28, 1994 **Accepted:** October 5, 1994

In this paper the rough set theory (RST) approach to machine learning is analysed and its drawbacks described. RST often uses complicated formalization of rather simple notions and sometimes invents new notions that make the RST papers hard to read and understand. Some authors from the RST community tend to ignore the huge amount of work done in machine learning. This may lead to reinventings and ad-hoc solutions.

1 Introduction

Pawlak (1982) defined the rough sets theory (RST) which was later used for several applications and published in a series of papers (e.g. Pawlak, 1984; Pawlak et al., 1986). However, Wong et al. (1986) compared the RST approach with ID3 approach (Quinlan, 1979; 1986) and concluded that the inductive learning defined by Pawlak is just a special case of the ID3 approach and basically differs in some (unrealistic) assumptions. To overcome the deficiencies Wong & Ziarko (1986) defined the probabilistic RST which was later refined by Pawlak et al. (1988). The redefined probabilistic RST seems to eliminate some drawbacks of the original (discrete) RST, while still containing some problematic issues.

Later, numerous papers were published on the application of RST, which almost all use the original definition of (discrete) RST (with all its drawbacks). The most notable publications seem to be the paper (Kubat, 1991), the book by Pawlak (1991), and the edition of 27 papers in (Slowinski, 1992). Kubat ends his paper with "It is a pity that this topic (RST) has not yet received more publicity ...". Maybe the more appropriate claim would be: "It is a pity that nobody critically analysed the RST and compared the performance of RST with well known inductive learning approaches". The aim of this paper is to fill this gap.

In the next section the RST approach to machine learning is briefly described. We describe the basic terminology and definitions to give the reader the impression about the RST approach, however, in the RST literature there are many more definitions and terminology which often confuses the point. In section 3 we discuss the deficiencies of the RST approach: its formal and unreadable terminology, inflexible knowledge representation, and ad-hoc solutions. Section 4 describes experimental comparison of performance of two "classical" machine learning algorithms with the performance of RST. In conclusion we analyse the "contribution" of RST to machine learning.

2 Rough sets theory

2.1 Basic definitions

The (discrete) RST (Pawlak, 1982) is introduced through an *information system* which is defined as a 4-tuple $S =< U, Q, V, f >$, where U is a finite set of objects, Q is a finite set of attributes, V is the union of attributes domains and $f : U \times Q \rightarrow V$ is a total function that assigns a value to each attribute of each object. This function is used to define equivalence relation called *indiscernibility relation* for each subset of attributes $P \subset Q$:

$$\tilde{P} = \{(x,y) | x, y \in U \;\&\; \forall q \in P : f(x,q) = f(y,q)\} \tag{1}$$

In other words, objects x and y are in the relation $\tilde{P}$ if they have the same value for every attribute from P. The family of equivalence classes of relation $\tilde{P}$ is denoted by P^*.

A *rough set* is defined as an approximation of set of objects $Y \subset U$ and is defined with two unions of equivalence classes of objects (Pawlak, 1982):

$$\underline{P}Y = \bigcup\{X \in P^* | X \subseteq Y\} \tag{2}$$
$$\overline{P}Y = \bigcup\{X \in P^* | X \cap Y \neq \emptyset\}$$

$\underline{P}Y$ is called P-lower approximation of Y and $\overline{P}Y$ is called P-upper approximation of Y. The P-lower approximation therefore contains only objects from set Y while P-upper approximation contains also objects from $U - Y$ which are "P-indiscernible" from some objects from Y. The relation between defined sets and set Y is:

$$\underline{P}Y \subseteq Y \subseteq \overline{P}Y \tag{3}$$

An approximation is sometimes represented also in terms of three regions:

- P-positive region of set Y in S :

$$POS_P(Y) = \underline{P}Y \tag{4}$$

- P-negative region of set Y in S :

$$NEG_P(Y) = U - \overline{P}Y \tag{5}$$

- P-boundary (doubtful) region of set Y in S :

$$BND_P(Y) = \overline{P}Y - \underline{P}Y \tag{6}$$

P-positive and P-negative regions contain P-equivalence classes containing objects that all belong and all do not belong to set Y, respectively. P-doubtful region is the union of P-equivalence classes each of which contains some objects that belong to Y and some objects that do not.

Approximation of family of sets is a generalization of approximation of set Y. If χ is defined as:

$$\chi = \{Y_1, Y_2, \ldots, Y_m\} \quad : \quad Y_i \subseteq U \tag{7}$$

then approximation of χ by P is defined:

$$\begin{array}{l} \underline{P}\chi = \{\underline{P}Y_1, \underline{P}Y_2, \ldots, \underline{P}Y_m\} \\ \overline{P}\chi = \{\overline{P}Y_1, \overline{P}Y_2, \ldots, \overline{P}Y_m\} \end{array} \tag{8}$$

If χ is a *classification* (i.e. $Y_i \cap Y_j = \emptyset \; : \; \forall i, j \leq m\,,\; i \neq j,\; \bigcup_{i=1}^{m} Y_i = U$) then measures of *roughness* of approximation of χ are defined as follows:

- *accuracy of approximation of χ by P in S* or shortly *accuracy of classification χ*

$$\beta_P(\chi) = \frac{\sum_{i=1}^{m} card(\underline{P}Y_i)}{\sum_{i=1}^{m} card(\overline{P}Y_i)} \qquad 0 \leq \beta_P(\chi) \leq 1 \tag{9}$$

- *quality of approximation of classification of χ by P in S* or shortly *quality of classification χ*

$$\gamma_P(\chi) = \frac{\sum_{i=1}^{n} card(\underline{P}Y_i)}{card(U)} \qquad 0 \leq \gamma_P(\chi) \leq 1 \tag{10}$$

2.2 Decision rules

If *classification* $\chi = \{Y_1, Y_2, \ldots, Y_m\}$ is defined in *information system* $S =< U, Q, V, f >$ and a set of attributes $P \subseteq Q$ induce P-equivalent classes $X = (X_1, X_2, \ldots, X_n)$, decision rules for set Y_j are defined (Pawlak, 1991):

$$OP_j =$$
$$\{r_{ij} : (Des(X_i) \Rightarrow Des(Y_j)); X_i \cap Y_j \neq \emptyset; i = 1..n\}$$

and decision rules for a whole system are then defined:

$$OP(\chi) = \{OP_j\,;\, j = 1\,..\,m\} =$$
$$\{r_{ij} : (Des(X_i) \Rightarrow Des(Y_j))\,;\; X_i \cap Y_j \neq \emptyset\,;$$
$$i = 1\,..\,n\,;\, j = 1\,..\,m\} \tag{11}$$

where $Des(X_i)$ is a description of an equivalence class which is equivalent to the description of all objects from that class:

$$\forall x \in X_i : Des(x) =$$
$$(p_1 = v_{j_1}) \wedge (p_2 = v_{j_2}) \wedge \ldots \wedge (p_n = v_{j_n}) = Des(X_i) \tag{12}$$

where $x \in U$, $p_k \in P$, $v_{j_k} \in V_{p_k}$. $Des(Y_j)$ represents a description of the value of an *action* attribute (class).

Two types of rules are defined:

- r_{ij} is DETERMINISTIC $\Longleftrightarrow$ $X_i \cap Y_j = X_i$
- r_{ij} is NONDETERMINISTIC $\Longleftrightarrow$ $X_i \cap Y_j \neq X_i$

2.3 Learning algorithm

To generate a set of decision rules a subset P of attributes must be selected. Decision rules are generated using only attributes from P. For both these complex tasks the RST approach uses (ad-hoc) heuristics. There are some more definitions.

Let $S = (U, Q, V, f)$ be an information system.

- Set of attributes $R \subseteq Q$ *depends* on set of attributes $P \subseteq Q$ in S (denotation $P \to R$) $\Longleftrightarrow$ $\widetilde{P} \subseteq \widetilde{R}$.
- Set of attributes $P \subseteq Q$ is *independent* in S $\Longleftrightarrow$ $\forall P' : P' \subset P \Rightarrow \widetilde{P'} \supset \widetilde{P}$
- Set of attributes $P \subseteq Q$ is *dependent* in S $\Longleftrightarrow$ $\exists P' : P' \subset P \Rightarrow \widetilde{P'} = \widetilde{P}$ $\quad (P' \to P)$
- Set of attributes $P \subseteq Q$ is *reduct* in S $\Longleftrightarrow$ P is the greatest *independent* set in Q

The definition of a *reduct* is from (Pawlak et al., 1986). The "greatest independent set" is interpreted as locally greatest and not globally. In Slowinski (1992) a reduct is defined as the minimal set of attributes that define the same equivalence classes as original set of attributes (which is equivalent to the above definition). Here the word "minimal" is again interpreted as locally and not globally minimal.

From the above definitions we have the following properties:

1. If set of attributes $P \subseteq Q$ is *independent* in S $\Longrightarrow \forall p, q \in P : \neg(p \to q)$ & $\neg(q \to p)$
2. If set of attributes $P \subseteq Q$ is *independent* in S $\Longrightarrow \forall P' \subset P : card(P'^*) < card(P^*)$, $(\widetilde{P'} \supset \widetilde{P})$
3. Set of attributes $P \subseteq Q$ is *dependent* in S $\Longleftrightarrow$ $\exists P' \subset P$: P' is *independent* & $P' \to P - P'$

There may exist more than one reduct $P \subseteq Q$.

Learning algorithm as defined in RST, is divided into three steps:

1. Reduction of those attributes from set Q that do not change the Q-equivalence classes. For such attribute p holds $(Q - p) \to Q$. This leads to *independent* set of attributes A for which stands:

   $$Q^* = A^* \qquad \gamma_Q(\chi) = \gamma_A(\chi)$$

   As the number of reducts may be large one should need a preference criterion for searching for "good reduct". In the RST literature typically an ad-hoc search is performed, such as try to eliminate first attribute, then second etc. (see e.g. Pawlak et al., 1986; Slowinski & Slowinski, 1990; Slowinski, 1992a; Tanaka et al., 1992; Grzymala-Busse, 1992).

2. Elimination of attributes from reduct A (which causes joining some of equivalence classes and therefore decreases the quality of classification γ) until the lowest permitted predefined value of γ is reached. The search heuristic in each step eliminates the attribute that maximizes γ for remaining subset P of attributes.

3. Generation of rules for each class in turn using only attributes from P. Many authors from the RST community use the covering algorithm (e.g. Wong et al., 1986; Wong & Ziarko, 1986). In each iteration one rule is generated and correctly classified training instances are removed. Iterations terminate when all training instances are removed or when no more rules can be found.

   One rule is generated in a top-down manner, starting with empty condition, and by specializing the current rule in all possible ways and selecting the most promising specialization. This continues until the certain quality criterion is met. For deterministic RST, the quality criterion is the determinism of the conclusion part of the rule (Wong et al., 1986). For probabilistic RST, the quality criterion requires that the percentage of the majority class is above the user defined threshold $\alpha \geq 0.5$ (Wong & Ziarko, 1986).

Classification with decision rules is straightforward. If the description of an object is equal to the description of the condition attributes of the rule, the object is classified in the class represented with the value of the action attribute of that rule. For nondeterministic rules the discrete RST doesn't give any priority to classes as it assumes equal distribution of classes Y_j in the equivalence class. That was improved in the probabilistic RST (Wong & Ziarko, 1986; Pawlak et al., 1988) where for the conclusion part the distribution of covered training instances was used.

Another problem is if the rule for classification does not exist (i.e. there is no rule with the description of condition attributes corresponding to the description of an object). In this case the algorithm searches for k closest rules that best match the object and their conclusion part is averaged (Krusinska et al., 1992).

3 Drawbacks of RST

3.1 Terminology

Probably the most confusing thing with RST is its complicated formalization of rather trivial (at least in the context of machine learning from examples) notions. While formalization is usually welcome and necessary to avoid confusion, in RST it adds confusion with numerous new notions and unusual terminology that often conflicts with the usual terminology of machine learning community.

The definition of boundary region BND_P, which is the central part of the RST, merely represents a part of the instance space where attributes do not suffice for discriminating between classes. The whole concept of a rough set merely states that certain classes cannot be discriminated from other classes. However, there is no information provided about the distribution of instances inside the boundary region.

The accuracy of classification β, defined with eq. (9), should not be confused with the usual meaning of classification accuracy. The quality of classification γ, defined with eq. (10), in fact represents the percentage of training objects that can be correctly classified using only attributes from P. This is equal to the classification accuracy on training data which may of course drastically differ from the classification accuracy on unseen objects. Therefore, γ is poor search heuristic and the same holds for the quality criterion based on the majority class (see step 2 and 3 of the learning algorithm in section 2.3).

The definition of *dependent* and *independent* set of attributes is unusual in the context of machine learning. It uses only logical dependency into acount which is not enough. (In)dependency is defined in the probability theory, however, RST doesn't operate with probabilities at all. Pawlak et al. (1988) try to overcome this (but only for probabilistic RST) by defining the measure of dependency in terms of entropy measure. However, they define that attributes (variables) X and Y are completely independent if

$$H(Y|X) = \log m$$

where m is the number of values of attribute Y. It is well known that $H(Y|X) \leq H(Y)$ and that X and Y are independent only when $H(Y|X) = H(Y)$, and $H(Y)$ is only in a very special (and rare) case equal to $\log m$.

3.2 Knowledge representation

The derived rules use a fixed subset of attributes and discard probably useful information contained in other attributes. The authors of RST claim that such attributes are redundant and unnecessary. This may be true for noise-free, complete data sets with exact classification, which obviously is not the case for the great majority of classification problems. For a certain problem subspace one subset of attributes may be relevant and for the other subspace another subset of attributes may be crucial.

On the other hand, there is no notion of the probability distribution and the reliability of conclusion parts of decision rules. Deterministic decision rules, which are in fact just a special case of nondeterministic rules, are supported from n training instances belonging to same class where $n \geq 1$. Obviously, for small values of n, the conclusion becomes unreliable and probability distribution should be estimated using Laplace's law of succession and m-estimate (Cestnik, 1990). Although the probabilistic RST is more flexible with this respect, almost all authors use the deterministic RST.

RST can deal with discrete attributes only and continuous attributes have to be discretized in advance. There is no obvious way how to deal with incomplete data (missing values) and noisy data. The only straightforward solution to the problem of missing values is to define the unknown value as an additional value of an attribute, which is known to be unsatisfactory (Quinlan, 1989).

Some authors from the RST community use a *preset decision tree* for the knowledge representation (see e.g. Modrzejewski, 1993). A preset decision tree is a decision tree that has the same order of attributes for all paths from the root to the leaf. This has an obvious disadvantage to be less flexible than ordinary decision trees which are in turn less flexible than decision rules (Quinlan, 1987).

Kubat (1991) describes an algorithm for updating lower approximation $\underline{P}Y$ of concept Y and upper approximation $\overline{P}Y$ using the fixed set of attributes P. Both approximations are represented simply as sets of equivalence classes of relation $\widetilde{P}$. The equivalence classes contain objects with the same values for all attributes. Equivalence classes are disjoint and the union of all equivalence classes is equal to the set of all training instances. Theorems in the paper by Kubat are more or less trivial, although the awkward formal description is clumsy for reading and understanding. They simply describe how new objects can be added to these sets and how existing objects can be deleted. The only learning in this framework is therefore the memorization. There is no induction, no generalization and neither specialization. The knowledge is represented in the same way as it was provided to the learner, except that objects are grouped into the equivalence classes. Equivalence classes containing objects from more than one class are in the boundary region $BND(Y)$. Obviously, such framework is not very useful and, besides, it is drastically sensitive to noise and missing data.

3.3 Ad-hoc solutions

Instead of using well known results from the probability theory and the information theory, the authors from the RST community often use ad-hoc definitions and solutions. There is plenty of parameters and thresholds with poor theoretical background. While dealing with definitions of straightforward notions, the RST is strictly formal and rigorous. As soon as more interesting problems are encountered, such as finding a good reduct or searching for good rules, ad-hoc heuristics are used.

Such ad-hoc heuristics are described in section 2.3 (learning algorithm) and discussed in section 3.1 above. Note, that the definition of a *reduct* is ad-hoc, at least in the context of machine learning, as it is not connected to the class attribute at all. Having enough random attributes, a huge number of reducts can be found that does not reflect any domain regularities at all. Therefore, any definition of a "good" reduct is also ad-hoc and any search heuristic for finding a reduct is necessarily ad-hoc.

3.4 Comparisons of RST to other approaches

Although there are many applications of RST (see e.g. Slowinski, 1992), there was practically no comparison of performance with existing machine learning algorithms. Babič et al. (1992) compared the performance of Assistant (Cestnik et al., 1987) with CART (Breiman et al., 1984) on two medical data sets. They reported 74% and 84% of classification accuracy, for each data set in turn, achieved by Assistant. The same authors (Krusinska et al., 1992) tested the performance of the RST learning algorithm on the same data sets and reported 73% and 80% of classification accuracy for each data set in turn They also report about results, obtained by Assistant Professional package, but only when using unusual classification methods in combination with the naive Bayesian classifier which does not use m-estimate of probabilities (Cestnik et al, 1987); and these results were worse than results reported in (Babič et al., 1992).

Wong et al. (1986) theoretically analysed RST and ID3 (Quinlan, 1979) and concluded:

"The criterion for selecting dominant attributes based on the concept of rough sets is a special case of the statistical method if equally probable distribution of objects in the doubtful region of the approximation space is assumed."

The assumption of equally probable (uniform) distribution is far from realistic.

Teghem & Benjelloun (1992) performed similar analysis and concluded in the self contradictory statements:

"If RST certainly is an efficient tool to analyse

information systems, often more simple and comprehensive than Quinlan's method using entropy notion, nevertheless the comparison suggests that some improvements can still be effected in the rough sets approach. Further researchs are necessary to investigate how the distribution of the objects into doubtful regions can be taken into account."

4 Experimental comparison

We reimplemented the RST learning algorithm and tested its performance on several real world data sets. We compared the performance with "classic" machine learning algorithms: Assistant inductive learning algorithm for generating decision trees (Cestnik et al., 1987) and the naive Bayesian classifier with m-estimate of probabilities (Cestnik, 1990).

The important characteristics of our implementation of RST learning algorithm are:

- We used the RST learning algorithm, described by Wong et al. (1986) generalized to probabilistic RST (Wong & Ziarko, 1986; Pawlak et al., 1988). We tried different values of majority class limit α. Here we present the best results obtained for each data set. This is somehow an overestimation of the performance of the RST learning algorithm.
- In the case of searching for k "closest rules" to the testing object, k was set to 1 and the distance between rules and instances was defined as the number of condition attributes that have different value.
- Unknown value was treated as an additional value of the attribute. Namely, RST does not provide any methodology to deal with this problem.

The description of data sets used in our experiments is provided in table 1. Besides the usual numeric description of the data (number of attributes, number of classes, and number of cases) we provide also the class entopy and the proportion of the cases from the majority class. The class entropy shows how simple/hard is the classification problem while the proportion of the majority class shows the "default accuracy", i.e. the accuracy that can be achieved with a simple classifier that classifies all instances in the majority class.

One experiment (trial) consisted of dividing the set of objects into 70% for learning and 30% for testing. We performed 10 experiments, each with different split, and results were averaged. The measured parameters were:

- classification accuracy (the percentage of correctly classified testing instances), results are presented in table 2;
- average information score, a measure that eliminates influence of prior probabilities and is defined as follows (Kononenko & Bratko, 1991):

$$Inf = \frac{\sum_{i=1}^{\#testing\ instances} Inf_i}{\#testing\ instances} \qquad (13)$$

where the information score of classification of i-th testing instance is defined with the following. Let Cl_i be the class of i-th testing instance, $P(Cl)$ the prior probability of class Cl and $P'(Cl)$ the probability returned by a classifier. We define the information score for two cases:

The information score is positive if the probability of the correct class given by the classifier is greater than the prior probability of that class. The information gain is equal to the prior information minus the posterior information necessary to correctly classify that instance:

$P'(Cl_i) \geq P(Cl_i)$:

$$Inf_i = -log_2 P(Cl_i) + log_2 P'(Cl_i)$$

If the classifier decreases the prior probability of the correct class, the provided information is wrong and therefore negative. It is equal to the prior information minus the posterior information necessary to incorrectly classify that instance:

$P'(Cl_i) < P(Cl_i)$:

$$Inf_i =$$

$$-(-log_2(1 - P(Cl_i)) + log_2(1 - P'(Cl_i)))$$

Results are presented in table 3.

Table 1: Characteristic description of experimental data sets.

domain name	# attributes	# classes	# cases	class entropy	majority class
primary tumor	17	22	339	3.64bits	25%
breast cancer	10	2	288	0.72bits	80%
thyroid diseases	15	4	884	1.59bits	56%
rheumatology	32	6	355	1.70bits	66%
hepatitis	20	2	155	0.73bits	79%
lymphography	19	4	148	1.23bits	55%
criminology	11	4	723	1.34bits	64%
fresh concrete	14	4	254	1.77bits	42%

Table 2: Comparison of the classification accuracy (%) of different classifiers on various data sets.

domain name	Assistant	naive Bayes	RST
primary tumor	44	50	35
breast cancer	77	79	80
thyroid diseases	73	72	61
rheumatology	65	69	66
hepatitis	82	87	81
lymphography	79	84	77
criminology	61	61	63
fresh concrete	61	63	61

Table 3: Comparison of the average information score (bit) of different classifiers on various data sets.

domain name	Assistant	naive Bayes	RST
primary tumor	1.38	1.57	0.96
breast cancer	0.07	0.18	-0.04
thyroid diseases	0.87	0.85	0.46
rheumatology	0.46	0.58	0.16
hepatitis	0.15	0.42	0.12
lymphography	0.67	0.83	0.51
criminology	0.06	0.27	0.03
fresh concrete	0.70	0.89	0.59

All differences in the classification accuracy (table 2) that are less than 4 % are statistically insignificant (confidence level is 0.99 using two-tailed t-test). Other differences are significant. However, note that for breast cancer, rheumatology, and criminology, where the differences are the lowest, the classification accuracy is practically equal to the proportion of the majority class. For those data sets the information score is a better measure. The majority of differences in information score (table 3) are statistically significant (the exceptions are the differences between Assistant and RST in hepatitis and criminology).

Results of RST are poor when compared to Assistant and the naive Bayesian classifier with respect to classification accuracy and/or information score. Therefore, RST did not achieve the performance of "classic" machine learning algorithms. Besides, nowadays there exist better machine learning algorithms which usually obtain better performance with multistrategy learning (e.g. Quinlan, 1993; Brodley, 1993). In fact we used a multistrategy approach (in the sense of multiple sets of rules) in our implementation of RST to improve the performance of RST. However, the underlying assumptions prevented the RST learning algorithm to perform well.

5 Conclusion

Lists of references in the papers on RST contain plenty of self referencing while none or modest number of references from machine learning literature. There are some exceptions (e.g. Grzymala-Busse, 1992), however the only purpose of referencing in such cases is to inform about alternative approaches without any explicit comparison. It seems that many authors have no overview of the work that is going on in machine learning and that may be the reason for many reinventings and also plenty of ad-hoc solutions.

Complicated formalization in RST adds confusion with numerous new notions and unusual terminology that prevents global overview of the RST and prevents systematic analysis. This may be why so many authors use RST without analysing its basic assumptions, which are in most cases unrealistic. The problems with noise and incomplete data disable RST from providing efficient solutions for complex real-world problems.

Acknowledgements

We are grateful to Ankica Babič for providing the literature on RST and Matevž Kovačič and Uroš Pompe for their comments on the manuscript.

References

[1] Babič. A., Krusinska E., Stromberg J.E. (1992) Extraction of diagnostic rules using recursive partitioning systems: A comparison of two approaches. *Artificial Intelligence in Medicine*, 4: 373-387.

[2] Breiman L., Friedman J.H., Olshen R.A., Stone C.J. (1984) *Classification and Regression Trees*, Wadsforth International Group.

[3] Brodley C.E. (1993) Addressing the selective superiority problem: automatic algorithm/model class selection. *Proc. 10th Int. Conf. on Machine Learning*, (Amherst, MA, June 1993), Morgan Kaufmann, pp. 17-24.

[4] Cestnik B. (1990) Estimating probabilities: A crucial task in machine learning, *Proc. European Conf. on Artificial Intelligence*, Stockholm, August, 1990, pp. 147-149.

[5] Cestnik B., Kononenko I.& Bratko I. (1987) ASSISTANT 86 : A knowledge elicitation tool for sophisticated users, in: I.Bratko, N.Lavrač (eds.): *Progress in Machine learning*, Wilmslow: Sigma Press.

[6] Grzymala-Busse J.W. (1992) LERS - A system for learning from examples based on rough sets. In. Slowinski R. (ed.) *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publ.

[7] Kononenko I. & Bratko I. (1991) Information based evaluation criterion for classifier's performance, *Machine Learning*, 6: 67-80.

[8] Krusinska E., Babič A., Slowinski R., Stefanowski J. (1992) Comparison of the rough sets approach and probabilistic data analysis techniques on a common set of medical data. In. Slowinski R. (ed.) *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publ.

[9] Kubat M. (1991) Conceptual inductive learning: the case of unreliable teachers. *Artificial Intelligence*, 52: 169-182.

[10] Modrzejewski M. (1993) Feature selection using rough sets theory. *Proc. European Conf. on Machine Learning*, Ed. P. Brazdil, Springer Verlag, pp.213-226.

[11] Pawlak Z. (1982) Rough sets. *Int. J. of Computers and Information Sciences*, Vol. 11, pp. 341-356.

[12] Pawlak Z. (1984) On superfluous attributes in knowledge representation. Bulletin of the Pollish Academy of Sciences, 32.

[13] Pawlak Z. (1991) *Rough Sets: Theoretical Aspects of Reasoning about Data.* Kluwer Academic Publ.

[14] Pawlak Z., Slowinski K., Slowinski R. (1986) Rough classification of patients after highly selective vagotomy for duodenal ulcer. *Int. J. Man-Machine Studies*, 24: 413-433.

[15] Pawlak Z., Wong S.K.M., Ziarko W. (1988) Rough sets: probabilistic versus deterministic approach. *Int. J. Man-Machine Studies*, 29: 81-95.

[16] Quinlan, J.R. (1979) Discovering Rules by Induction from Large Collections of Examples. In: D.Michie (ed.), *Expert Systems in the Microelectronic Age.* Edinburgh University Press.

[17] Quinlan, J.R. (1986) Induction of Decision Trees. *Machine Learning.* 1: 81-106.

[18] Quinlan J.R. (1987) Generating production rules from decision trees. *Proc. IJCAI-89*, Milan, August 1987, pp.304-307.

[19] Quinlan J.R. (1989) Unknown attribute values in induction, *Proc. 6th Int. Workshop on Machine Learning*, Cornell University, Ithaca, June 26-27, 1989, pp.164-168.

[20] Quinlan J.R. (1993) Combining instance-based and model-based learning. *Proc. 10th Int. Conf. on Machine Learning*, (Amherst, MA, June 1993), Morgan Kaufmann, pp. 236-243.

[21] Slowinski K. (1992a) Rough classification of HSV patients. In. Slowinski R. (ed.) *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publ.

[22] Slowinski K. & Slowinski R. (1990) Sensitivity analysis of rough classification. *Int. J. Man-Machine Studies*, 32: 693-705.

[23] Slowinski R. (ed.) (1992) *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publ.

[24] Tanaka H., Ishibuchi H. & Shigenaga T. (1992) Fuzzy inference system based on rough sets and its application to medical diagnosis. In. Slowinski R. (ed.) *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publ.

[25] Teghem J. & Benjelloun M. (1992) Some experiments to compare rough sets theory and ordinal statistical methods. In. Slowinski R. (ed.) *Intelligent Decision Support: Handbook of Applications and Advances of the Rough Sets Theory*, Kluwer Academic Publ.

[26] Wong S.K.M. & Ziarko W. (1986) INFER - An adaptive decision support system based on the probabilistic approximate classification.*Proc. 6th Int. Workshop on Expert Systems and their Applications*, Avignon, France, pp. 713-726.

[27] Wong S.K.M., Ziarko W., Li Ye R. (1986) Comparison of rough-set and statistical methods in inductive learning, *Int. J. Man-Machine Studies*, 24: 53-72.