To Combine or not Combine: Integration of Rough Sets in Multimethod Approach?

Matej Mertik, Gregor Štiglic, Mitja Lenič, Peter Kokol
Laboratory for System Design, University of Maribor, FERI
Smetanova 17, 2000 Maribor, Slovenia
matej.mertik@uni-mb.si

ABSTRACT

Due to the weakness of single methods their integration into hybrids is intensively researched in recent years. Most of these approaches use the static approach which is not successful in every situation therefore we build Multimethod approach that dynamically combines different machine learning methods following the assumption that only the synergetic combination of single models can unleash their full power. To increase the power of Multimethod approach we have to use as many as possible methods, which have to be integrated in Multimethod framework. In this paper we present a feasibility study for integration of the rough sets and its possible contributions to the system as a whole.

1. Introduction

The aggressive rate of growth of disk storage and thus the ability to store enormous quantities of data has far outpaced our ability to process and utilize that. This challenge has produced a phenomenon called data tombs – data is deposited to merely rest in peace, never to be accessed again. But the growing appreciation that data tombs represent missed opportunities in for example supporting scientific discovering, business exploitation or complex decision making has awaken the growing commercial interest in knowledge discovery and data mining techniques.

The appearance of new computer-based information technology and especially the introduction of intelligent systems with their ability to learn can enormously ease and improve these activities. Similar to mechanical systems that increase our physical abilities (cranes to lift vast amounts, telescopes to see farther, etc.), intelligent systems are power tools for heavy lifting in the information world - they complement, extend, and amplify our ability to think and solve problems.

However, selection of the machine learning method as a part of intelligent system strongly depends on the nature of the problem. There are many classical machine learning approaches, like decision trees, rules, rough-sets, case based reasoning, neural networks, support vector machines, different fuzzy methodologies, but they all have some advantages and limitations. There are not general rules, which describe the use of the appropriate method considering the problem. To achieve useful result considered a number of experiments with strategies using different approaches are required. In this paper we present our Multimethod approach, which is a powerful and promising technique based on an idea of a population of different intelligent systems that can produce multiple comparable good solutions, and a case study of the novel intelligent method inclusion. As a feasibility study we consider Rough Sets classification method, which represents knowledge in different form, and may therefore be suitable to improve Multimethod approach.

2. Multimethod approach

Historically different approaches for knowledge extraction evolved [1], such as symbolic approaches and computational learning theory. Among them we can find many classical approaches, like decision trees, rules, rough-sets, case based reasoning, neural networks, support vector machines, different fuzzy methodologies, ensemble methods [2], but they all have some advantages and limitations. Evolutionary approaches (EA) are also a good alternative, because they are not inherently limited to local solutions [3]. Recently, taking into account the limitations of classical approaches many researchers focused their research on hybrid approaches, following the assumption that only the synergetic combination of single models can unleash their full power [4].

Current studies show that the selection of appropriate method for data analysis can be crucial for the success. Therefore, for a given problem, different methods should be tried to increase the quality of extracted knowledge.

According to the previous paragraph a logical step would also be to combine different methods into one more complex methodology in order to overcome the limitations of a single method. We noticed that almost all attempts to combine different methods use loose coupling approach where the methods work almost independent of each other. Therefore a lot of "luck" and experiments with many different combinations are needed to unify them into a successful "team". Thus we decided to design a new approach that enables tight tangling of single methods. This new approach is called a Multimethod approach [5]. Opposed to the conventional hybrids our idea is to dynamically combine and apply different methods in not predefined order in the manner to solve a single problem or de-composition of that problem.

3. Knowledge sharing

Multimethod approach introduces the idea of a population of different intelligent systems that can produce multiple comparable good solutions, which are incrementally improved using the EA approach. In order to enable knowledge sharing between different methods the support for transformation between each individual method is provided. Initial population of intelligent systems is generated using different methods. In each generation different operations appropriate for individual knowledge representation are applied to improve existing and also to create new intelligent systems. That enables incremental refinement of extracted knowledge, with different aspects of a given problem. For example, using different induction methods such as different purity measures can be simply combined into a decision tree. As long as the knowledge representation is the same, a combination of different methods is not a big obstacle. The main problem is how to combine methods that use different knowledge representations (for example neural networks and decision trees). In such cases we provide two alternatives: (1) to convert one knowledge representation into another, using different already known methods or (2) to combine both knowledge representations into a single intelligent system. The first alternative requires implementation of knowledge transmutators (for example conversion of a neural network into a decision tree). Such conversions are not perfect and some of the knowledge is normally lost, but conversions can produce different aspect on the presented problem that can lead to better results.

The second alternative requires some cut-points where knowledge representations can be merged. In a decision tree internal nodes or decision leafs represent such cut points (Fig.1), i.e. a condition can be replaced by another intelligent system (for ex-ample support vector machine - SVM). We call such trees hybrid decision trees.

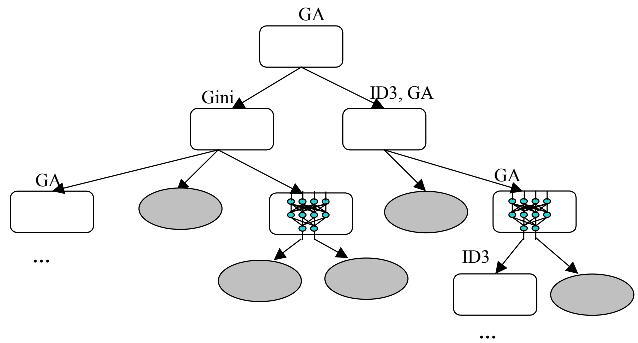

Fig.1. An example of a hybrid decision tree induced by the multimethod approach. Each node is induced with appropriate method (GA – genetic algorithm, ID3, Gini, SVM, neural network, etc.)

4. Method integration requirements

To successfully integrate new method in the Multimethod framework, such method has to have specific features. Most important one is to be able to share different aspects of a problem with another methods. Therefore it should have different learning and inference algorithm what implies intermediate knowledge representation. For example methods that use all learning instances such as Naïve Bayes cannot contribute a lot to Multimethod framework, but can use result of other methods that reduce instance space to reduce noise and improve reliability.

Key element and first requirement to new method is its intermediate knowledge representation. It is starting point, which can enable sharing of knowledge (hypothesis) and its aspects on a problem with other methods, that in cases where knowledge representation is different of existing ones implies knowledge transformation in different knowledge representations. However, conversion between different methods (e.g. decision trees and neural networks) is not perfect and some of the knowledge is normally lost, but what can lead to better results is the fact that conversion can produce a different aspect on a presented problem. Also merged knowledge representations in our hybrid decision trees where the cut-point (be i.e. condition) can be replaced by another intelligent system, assure us with different aspect of the presented problem that can lead to better solutions as discussed in 6. But in general the methods which want to share the knowledge in Multimethod approach need to

satisfy a list of requirements which (1) enables an integration of a method itself into Multimethod framework and (2) the characteristic for method's transformation.
The first requirement considers aspect of implementation. The idea of Multimethod approach is implemented in tool MultiVeDec (Multimethod Vector Decision) that was developed in Java environment. To integrate a novel method into MultiVeDec tool we must implement several interfaces, which are used for communication between different intelligent systems that can produce multiple comparable good solutions. It is not intention of the paper to cover the programmatic aspect of the framework, so we will not go more deeply into the technical details. However readers, which are interested in it, can found more in [5].
The second requirement considers knowledge conversion (for example conversion of a neural network into a decision tree). Such conversions can produce a different aspect on a presented problem that can lead to better results. However they are not perfect and some of the knowledge is normally lost. The method that is candidate for the integration into Multimethod framework must be considered from the aspect of knowledge conversion.
MultiVeDec was primarily developed for application in medical field and therefore its final knowledge representation should be in symbolic form. For that reason it is desirable that new method has the ability to convert its knowledge in that form.
In the following paragraph we will evaluate Rough Sets as a candidate method for inclusion in Multimethod framework.

5. Rough Sets

Rough Sets theory constitutes a framework for inducing minimal decision rules. These rules are then used for classification tasks. The goal of this technique is to search large databases for meaningful decision rules and finally acquire new knowledge. It is relatively new technique in the field of mathematics and artificial intelligence, which was introduced in the early eighties by a Polish mathematician Zdizislaw Pawlak [6]. It was one of the less known techniques for the most of the eighties, but became popular in the middle of nineties.
The original concept behind the Rough Sets theory is the realization that sets can be described “roughly” i.e. there are three regions of knowledge. This means that an object can have a property “certainly”, “possibly” and “certainly not” (Fig.2.).

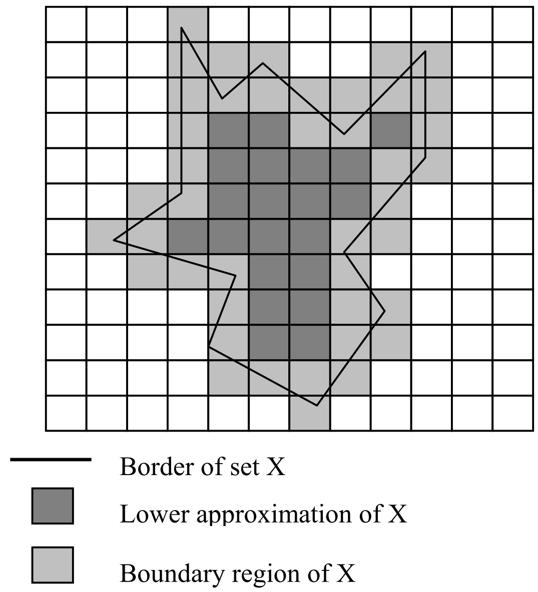

Fig.2. Simple representation of Rough Sets Method (three regions of knowledge)

Rough set analysis uses only internal knowledge, and does not rely on prior model assumptions as fuzzy set methods or probabilistic models do. In other words, instead of using external numbers or other additional parameters, rough set analysis uses only given data. We can also look at a Rough sets method as a “black box” that is fed by training data and it produces a set of certain and uncertain rules. The following characteristics that satisfied our general methods requirements described in 4:
(1) The Rough Sets method is capable of acquiring new knowledge and is implemented in Java environment. It implements basic interfaces for black box inclusion in MultiVeDec.
(2) Results from the Rough Sets method are presented as minimal decision rules, which can be easily converted into a decision tree.
Rough set analysis can be used in a wide variety of disciplines, everywhere large amounts of data are being produced rough sets can be useful. One such field is medicine. Because many variables have an effect on the patient and because of large amounts of data these experiments bring forth, the human eye is bound to oversee details that can be brought to light by means of rough sets. We did some experiments on medical and other databases from UCI repository with the Rough Sets as is discussed further.

6. Results

For the empirical analysis of machine learning algorithms we used databases from the UCI Repository Of Machine Learning Databases [7]. The databases that we used in our testing were mostly from the medical field, however we also used two databases from other scientific areas. Information about number of instances, features and decision classes of databases is presented in Table 1.

Database	Num. of Instances	Num. of Features	Decision Classes
Pima	768	8	2
Hepatitis	155	19	2
Breast Cancer	699	9	2
Heart	270	13	2
Ionosphere	351	34	2
Vehicle	846	18	4

Table 1. Datasets description

As we presented in 5, Rough set analysis can be used in a wide variety of disciplines. In our experiment the databases contain a large number of attributes. Regarding Rough Set's ability of working with databases containing many features, we considered that we could get appropriate results from above mentioned databases.
We used simple voting as the classification strategy and equal frequency discretization for Rough Sets implementation. The reason for this is that simple voting usually gives the best results when the discretized attribute values are used. The results of average class and overall classification accuracy of Rough Sets method are presented in Table 2.

Database	Average Accuracy	Average Class Accuracy
Pima	66.27	67.80
Hepatitis	79.33	39.67
Breast Cancer	96.52	96.18
Heart	80.00	85.08
Ionosphere	86.65	87.22
Vehicle	58.69	60.19

Table 2. Rough Sets results on the UCI repository databases

We can see that our method had some problems classifying objects from databases where we have majority of objects in one class (Hepatitis database). It is also known that Rough Sets can be used for feature selection where they can achieve very good results [8, 9]. Based on that, we can assume that Rough Sets can help us find useful attributes that can then be used at building a tree in Multimethod approach. Moreover collaboration of Multimethod's participating methods that are sharing the knowledge together can lead us to better results.
To show Rough Sets ability to select features, we run 10-fold cross-validation tests on the same databases measuring effective feature reduction of the method. The results are presented in Table 3. We compare accuracy of ID3 algorithm on original dataset and on dataset with reduced features using Rough Sets feature selection.

Dataset	ID3 Accuracy	Num. Of Features	ID3 + FS Acc.	Num. of Features
Pima	70.53	8.0	66.71	4.2
Hepatitis	79.62	14.3	83.31	9.3
Breast	93.77	7.6	93.04	4.2
Heart	75.55	11.4	78.15	3.8
Ionosphere	86.86	12.5	84.57	13.1
Vehicle	73.69	17.5	70.48	13.0
Average	80.03	11.9	79.38	7.9

Table 3. Feature selection using Rough Sets method

We repeated tests on the same databases using Multimethod approach where we expected better results [Table 4]. Final knowledge representation was constrained to symbolic representation (only symbolic results were collected from final population), but not limited to it in evolution process.

Database	Average Accuracy	Average Class Accuracy
Pima	76.20	73.00
Hepatitis	82.70	65.90
Breast Cancer	97.00	97.87
Heart	86.70	87.40
Ionosphere	95.70	94.60
Vehicle	64.50	66.10

Table 4. Multimethod's approach results on UCI repository databases

Dynamical combining and application of different methods produced much better results than single Rough Sets method that were presented in Table 2. That confirms the idea of synergetic combination of methods, but does not necessarily mean that we cannot improve those results even further.

7. Improving Rough Sets for further experiments in Multimethod

Knowledge sharing combines different method approaches with transformation support for each individual method. As we determined for the Rough Sets in 4 they have a characteristic that enables the Rough Sets to be included in the Multimethod framework:
(1) Based on result in Table 2, the Rough Sets method is capable of acquiring new knowledge
(2) Results from the Rough Sets method are presented as minimal decision rules, which can be easily converted into decision trees.
After the analysis that we made, we concluded that the Rough Sets method is appropriate candidate for Multimethod inclusion. However, there are some improvements that should be done. Especially the second characteristic must be well considered, because the support for transformation of knowledge between different methods is crucial in the Multimethod framework. On the other hand there are also some improvements that can be done in Rough Sets method itself. We should improve its time complexity by implementing some of the techniques for faster feature selection.

8. Conclusion

Benefits of including new methods in Multimethod framework are quite obvious. Resulting synergetic combination and its collaboration with sharing different aspects can improve quality of produced intelligent systems. On other hand integrating new method presents additional costs that have to be considered.
The described approach is in the phase of integrating. We are predicting that in some cases the Rough Sets approach in Multimethod framework can produce better results like Rough Sets itself because of Multimethod's participating methods, which are collaborating together by knowledge sharing. On the other hand we are also aware of facts that Multimethod framework does not guarantee better results in every experiment because of theoretical principles, but enables the possibility that the methods together in some cases can produce better results. However in this paper we presented arguments for inclusion and against it. Based on the experiments, we concluded that the Rough Sets method is appropriate candidate for inclusion, not in task of classification and knowledge extraction, but in task of feature selection, as used by different other authors [6,9] Rough set method has to be additionally improved with more complex equivalent relations. After that it can be used in Multimethod framework as appropriate and reliable framework member.

References

1. Thrun, S., Pratt, L.: Learning to Learn. Kluwer Academic Publishers (1998)
2. Dietterich, T.G.: Ensemble Methods in Machine Learning. In: First International Workshop on Multiple Classifier Systems, Lecture Notes in Computer Science, Springer-Verlag, (2000), 1-15
3. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addi-son Wesley, Reading MA (1989)
4. Iglesias, C.J.: The Role of Hybrid Systems in Intelligent Data Management: The Case of Fuzzy/neural Hybrids, Control Engineering Practice, Vol. 4, No. 6 (1996) 839-845
5. Lenič, M., Kokol, P.: Combining classifiers with multimethod approach. In Soft computing systems: design, management and applications, IOS Press, (2002) 374-383
6. Pawlak, Z., Grzymala-Busse, J., Slowinski, R., Ziarko, W.: Rough Sets. Communications of the ACM, vol. 38, no. 11 (1995) 89-95
7. Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases [http://www.ics.uci.edu/~mlearn/MLRepository.htm]. Irvine, CA: University of California, Department of Information and Computer Science.
8. Modrzejewski, M.: Feature selection using rough sets theory. In P. Brazdil (Ed.), Proceedings of the European Conference on Machine Learning, (1993) 213-226
9. Štiglic, G., Lenič, M., Kokol, P.: Rough sets based feature selection. In M.H. Hamza (Ed.), Proceedings of the Third IASTED International Conference on Artificial Inteligence and Applications, (2003) 430-433