Predicting Students' Results Using Rough Sets Theory

Anca Udristoiu, Stefan Udristoiu, and Elvira Popescu

University of Craiova, Faculty of Automation, Computers and Electronics
Bvd. Decebal, Nr. 107, 200440, Craiova, Dolj, Romania
{anca_soimu,s_udristoiu,elvira_popescu}@yahoo.com

Abstract. This paper proposes the utilization of rough set theory for predicting student scholar performance. The rough set theory is a powerful approach that permits the searching for patterns in e-learning database using the minimal length principles. Searching for models with small size is performed by means of many different kinds of reducts that generate the decision rules capable for identifying the final student grade.

Keywords: education data mining, rough set theory, decision rules, learning management system.

1 Introduction

Predicting students' performance is one of the most important and useful applications of educational data mining and its goal is to score or mark from student course behavior and activity [14]. In this study, the rough set technique is used to offer methods for understanding, processing and modelling data, resolving the limitations of the e-learning systems.

The rough set theory was discovered by Zdzislaw Pawlak [1] and is a powerful mathematical tool for modeling the imperfect and incomplete knowledge [2]. Rough set theory has also excellent results in approximate reasoning [5], mathematical logic analysis and reduct [6, 7, 8], building of predictive models [9], and decision support system [10, 11, 12]. Many studies have shown that the use of rough set theory formulate a clear decision-making projects and enhance the effectiveness of the research while doing optimization [7]. The research related to education of Qu and Wang [11] provided a basis of personalized teaching strategies in distance learning website by analysis of reduct and attribute significance. In [13] the study analyzed students' misconception based on rough set theory. Although the rough set theory is rarely used in education, in this study we use its characteristics, which are very suitable for discovering rules useful in educational process.

2 Student Representation and Discretization

We have collected data from on-line course activity provided by Moodle [15] that is one of the most widely used open source learning management system. In fact, we have used the following data based on student 'Database' course activity [14]:

E. Corchado et al. (Eds.): IDEAL 2014, LNCS 8669, pp. 336–343, 2014.

Nassignment – number of assignments taken; *Nquiz* - number of quiz taken; *Nquiz_p* - number of quiz passed; *Nquiz_f* - number of quiz failed; *Nmessages* - number of messages sent to the chat; *Nmessages_ap* - number of messages sent to the teacher; *Nposts* - number of messages sent to the forum; *Nread* - number or forum messages read; *Total_time_assignment*- total time spent on assignment; *Total_time_quiz* – total time used in quizzes; *Total_time_forum*- total time used in forum; *Mark*- final mark the student obtained in the course.

Since the data provided by Moodle are structured, they didn't necessitate preparation [14]. So, we directly discretise them, transforming numerical values into categorical ones for a good interpretation and understanding. We have used the manual method for discretising all attributes, so the teacher has to specify the cut off points. The mark descriptor has four values: *insufficient*, if value < 5; *average*, if value > 5 and < 7; *good* if value >7 and < 9; *excellent* if value > 9. The other attributes have the values: LOW, MEDIUM and HIGH [14].

A student is represented in Prolog by means of a term:

```
student(ListofDescriptors)
```

where the argument is a list of terms used to specify the student attributes.

The term used to specify the student attributes is of the form:

```
descriptor(DescriptorName,DescriptorValue)
```

The model representation of students is in the following example:

```
student([
descriptor(Nassignment,medium),descriptor(Nquiz,low),
descriptor(Nquiz_p,low),descriptor(Nquiz_f,high),
descriptor(Nmessages, medium), descriptor(Nmessages_ap,
medium),descriptor(Nposts,low),descriptor(Nread,low),
descriptor(Total_time_assignment,low),
descriptor(Total_time_quiz,low),
descriptor(Total_time_forum,low)]).
```

3 Modelling of Student Information Using Rough Sets

3.1 Rough Sets Foundations

Rough sets theory is an intelligent mathematical tool and it is based on the concept of approximation space [1], [3]. In rough sets theory, the notion of information system determines the knowledge representation system. In this section, we recall some basic definitions from literature [1, 2, 3].

Let U denote a finite non-empty set of objects (students) called the universe. Further, let A denote a finite non-empty set of attributes. Every attribute $a \in A$, there is a function a: U → V_a, where V_a is the set of all possible values of a, to be called the domain of a. A pair IS = (U, A) is an information system. Usually, the specification of an information system can be presented in tabular form. Each subset of attributes

$B \subseteq A$ determines a binary *B–indiscernibility* relation *IND(B)* consisting of pairs of objects indiscernible with respect to attributes from B like in (1):

$$\mathrm{IND(B)} = \{(x, y) \in U \times U : \forall a \in B, a(x) = a(y)\} \quad (1)$$

IND(B) is an equivalence relation and determines a partition of *U*, which is denoted by *U/IND(B)*. The set of objects indiscernible with an object $x \in U$ with respect to the attribute set, *B*, is denoted by $I_B(x)$ and is called *B–indiscernibility* class:

$$I_B(x) = \{y \in U : (x, y) \in \mathrm{IND(B)}\} \quad (2)$$

$$U/\mathrm{IND(B)} = \{I_B(x) : x \in U\} \quad (3)$$

Table 1. Student Information System

U	Nmessages	Nmessages_ap	Mark
R_1	medium	low	average
R_2	medium	low	average
R_3	medium	low	average
R_4	medium	high	good
R_5	high	high	good
R_6	medium	medium	average
R_7	medium	medium	average
R_8	medium	high	good
R_9	medium	high	good
R_{10}	high	low	good
R_{11}	high	low	average

It is said that a pair ASB = (U, IND(B)) is an approximation space for the information system IS=(U, A), where $B \subseteq A$. The information system from Table 1 represents the students enrolled into a course represented in terms of descriptors values, as described in Section 2. For simplicity we consider only two descriptors as attributes, namely the Nmessages and Nmessages_ap. So, our information systems is IS = (U, B), where U = {R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11} and B={Nmessages, Nmessages_ap}.

Let $W = \{w_1,\ldots,w_n\}$ be the elements of the approximation space $AS_B=(U, IND(B))$. We want to represent *X*, a subset of U, using attribute subset *B*. In general, *X* cannot be expressed exactly, because the set may include and exclude objects which are indistinguishable on the basis of attributes *B,* so we could define *X* using the lower and upper approximation.

The *B-lower approximation X,* $\underline{B}X$, is the union of all equivalence classes in *IND(B)* which are contained by the target set *X*. The lower approximation of *X* is called the positive region of *X* and is noted *POS(X)*.

$$\underline{B}X = \bigcup \{ w_i \mid w_i \subseteq X \} \tag{4}$$

The *B-upper approximation* $\overline{B}X$ is the union of all equivalence classes in *IND(B)* which have non-empty intersection with the target set *X*.

$$\overline{B}X = \bigcup \{ w_i \mid w_i \cap X \neq \varnothing \} \tag{5}$$

Example: Let $X = \{R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8\}$ be the subset of *U* that we wish to be represented by the attributes set *B={Nmessages, Nmessages_ap}*. We can approximate *X*, by computing its *B-lower approximation,* $\underline{B}X$ and *B-upper approximation,* $\overline{B}X$. So, $\underline{B}X$ *={{R1, R2, R3}, {R5}, {R6, R7}}* and $\overline{B}X$ *={{R1, R2, R3}, {R5}, {R6, R7}, {R4, R8, R9}}.* The tuple ($\underline{B}X$, $\overline{B}X$) composed of the lower and upper approximation is called a rough set; thus, a rough set is composed of two crisp sets, one representing a *lower boundary* of the target set *X*, and the other representing an *upper boundary* of the target set *X*. The accuracy of a rough set is defined as: cardinality($\underline{B}X$)/cardinality($\overline{B}X$). If the accuracy is equal to 1, then the approximation is perfect.

3.2 Dispensable Features, Reducts and Core

An important notion used in rough set theory is the decision table. Pawlak [1] gives also a formal definition of a decision table: an information system with distinguished conditional attributes and decision attribute is called a decision table. So, a tuple DT = (U, C, D), is a decision table. The attributes C = {Nmessages, Nmessages_ap} *are* called conditional attributes, instead D = {Mark} is called decision attribute. The classes *U/IND(C)* and *U/IND(D)* are called condition and decision classes, respectively. The *C-Positive* region of *D* is given by:

$$POS_C(D) = \bigcup_{X \in IND(D)} \underline{C}X \tag{6}$$

Let $c \in C$ a feature. It is said that *c* is dispensable in the decision table *DT*, if $POS_{C-\{c\}}(D) = POS_C(D)$; otherwise the feature *c* is called indispensable in *DT*. If *c* is an indispensable feature, deleting it from *DT* makes it to be inconsistent.

A set of features *R* in *C* is called a reduct, if *DT'= (U, R, D)* is independent and $POS_R(D)=POS_C(D)$. In other words, a reduct is the minimal feature subset preserving the above condition.

3.3 Producing Rules by Discernibility Matrix

We transform the decision table into discernibility matrix to compute the reducts. Let *DT = (U, C, D)* be the decision table, with $U = \{R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8, R_9, R_{10}, R_{11}\}$. By a discernibility matrix of *DT*, denoted *DM(T),* we will mean *nxn* matrix defined as:

$$m_{ij}^{a(R_i)} = \{(a \in C : a(R_i) \neq a(R_j)) \text{ and } (d(R_i) \neq d(R_j))\} \tag{7}$$

where *i, j=1,2,...,11*.

The items within each cell of the discernibility matrix, *DM(DT)* are aggregated disjunctively, and the individual cells are then aggregated conjunctively. To compute the reducts of the discernibility matrix we use the following theorems that demonstrate equivalence between reducts and prime implicants of suitable Boolean functions [2], [12]. For every object Ri $\in$ U, the following Boolean function is defined:

$$g_{R_i}(\text{Nmessages}, \text{Nmessages\_ap}) = \bigwedge_{R_j \in U} \left(\bigvee_{a \in m_{ij}} a \right) \tag{8}$$

The following conditions are equivalent [3]:

1. $\{a_{i1}, \ldots, a_{in}\}$ is a reduct for the object Ri, i = 1..n. and
2. $a_{i1} \wedge a_{i2} \wedge .. \wedge a_{in}$ is a prime implicant of the Boolean function g_{R_i}

On Boolean expression the absorption Boolean algebra rule is applied. The absorption law is an identity linking a pair of binary operations. For example: $a \vee (a \wedge b) = a \wedge (a \vee b) = a$

From the decision matrix we form a set of Boolean expressions, one expression for each row of the matrix.

For the *average* mark, we obtain the following rules based on the table reducts:

- (Nmessages=mediu $\vee$ Nmessages_ap=low) $\wedge$ (Nmessages_ap =low) $\wedge$ (Nmessages=high)
- (Nmessages_ap =medium) $\wedge$ (Nmessages=medium $\vee$ Nmessages_ap =medium)
- (Nmessages=high $\vee$ Nmessages_ap =low) $\wedge$ (Nmessages_ap =low)

For the *good* mark we obtain the following rules based on the table reducts:

- Nmessages_ap = high
- Nmessages=high $\vee$ Nmessages_ap =high
- Nmessages_ap = high $\wedge$ (Nmessages=high $\vee$ Nmessages_ap =low)

By applying the absorption rule on the prime implicants, the following rules are generated:

- Rule 1: Nmessages_ap =low $\wedge$ Nmessages=high →average
- Rule 2: Nmessages_ap =medium→average
- Rule 3: Nmessages_ap =low→average
- Rule 4: Nmessages_ap = high→good
- Rule 5: Nmessages=high $\vee$ Nmessages_ap =high→good

3.4 Evaluation of Decision Rules

Decision rules can be evaluated along at least two dimensions: performance (prediction) and explanatory features (description). The performance estimates how well the rules classify unevaluated students. The explanatory feature estimates how interpretable the rules are [2]. Let be our decision table *DT = (U, C, D).* We use the set-theoretical interpretation of rules. It links a rule to data sets from which the rule is discovered [2]. Using the cardinalities of sets, we obtain the 2×2 contingency table representing the quantitative information about the rule *if descriptors then mark.* Using the elements of the contingency table, we may define the support (s) and accuracy (a) of a decision rule by:

$$s(\text{rule}) = \text{cardinality}(\text{descriptorSet} \cap \text{markSet}) \tag{9}$$

$$a(\text{rule}) = \frac{\text{cardinality}(\text{descriptorSet} \cap \text{markSet})}{\text{cardinality}(\text{descriptorSet})} \tag{10}$$

where the set $descriptorSet \cap markSet$ is composed from student descriptors which have a certain *descriptorSet* and a certain *mark*.

The coverage(c) of a rule is defined by:

$$c(\text{rule}) = \frac{\text{cardinality}(\text{descriptorSet} \cap \text{markSet})}{\text{cardinality}(\text{markSet})} \tag{11}$$

For the generated Rule 2, the contingency Table 2 is obtained, where the descriptor *Nmessages_ap* is denoted by *D*. For the Rule 2, the support is 2, accuracy is 2/2 and coverage is 2/6. In [4], the study suggests that high accuracy and coverage are requirements of decision rules.

Table 2. Contingency Table Representing the Quantitative Information about the Rule 2

	mark = average	not(mark = average)	
D =medium	cardinality(D=medium and mark=average) = 2	cardinality(D=medium and not(mark=average)) = 0	cardinality(D =medium)=2
not(D=medium)	cardinality(not(D =medium) and mark=average)=4	cardinality(not(D =medium) and not(mark=average))=5	cardinality(not(D =medium))=9
	cardinality(mark=average) =6	cardinality (not(mark=average))= 5	cardinality(U)=11

4 Decision Rule Extraction Using Rough Sets Models and Experiments

In this paper we present the application of rough set to discover student rules between students' descriptors and mark categories. A rule is represented using a Prolog fact:

```
rule(Mark, Accuracy, Coverage, ListofStudentDescriptors)
```

where *Mark*, the head of the rule, is the mark category, *Accuracy* is the rule accuracy computed as in (10), *Coverage* is the rule coverage computed as in (11) the body of the rule, is composed by conjunctions of student descriptors, while *Mark*, the head of the rule, is the mark category.

Decision rules are generated from reducts as described in Section 3. The student classification algorithm based on the discovered rules can be resumed as:

- collect all the decision rules in a classifier,
- compute for each rule the support, accuracy and coverage,
- eliminate the rules with the support less than the minimum defined support,
- order the rules by accuracy, than by coverage,
- if a student matches more rules select the first one: a student matches a rule, if all the descriptors, which appear in the body of the rule, are included in the descriptors of the student.

In the experiments realized through this study, two databases are used for the learning and testing process. The database used to learn the correlations between student behaviour and marks, contains information about 40 students, each described by 11 descriptors. For each mark class, the following metrics: accuracy, specificity, and sensitivity. The counted results are presented in Table 3.

Table 3. Results recorded for different marks

Mark	Accuracy(%)	Sensitivity(%)	Specificity(%)
Good	98.3	97	73.1
Average	97.7	96.1	72.8
Excellent	97.9	96	72.8
Insufficient	96.3	95.2	71.7

5 Conclusion

In this study, a method based on rough set theory is proposed and developed to assist the teacher by doing the pre-evaluation of students during a course study. For establishing correlations with the mark, we experimented and selected some descriptors of the student activity in the Moodle system for a “Database” course. The results of experiments are very promising and show that the methods based on rough set theory are very useful for predicting the results of the student during a course activity. The Prolog language used for representation of students’ descriptors and rules makes a simple and flexible integration of our methods with other learning management systems.

In future work, it would be interesting to repeat the analysis using more data from different types of courses and also to select other student descriptors. It would be also very useful to do experiments using more experts in order to analyse the obtained rules for discovering interesting relashionships.

Acknowledgment. This work was partially supported by the grant number 16C/2014, awarded in the internal grant competition of the University of Craiova.

References

1. Pawlak, Z., Skowron, A.: Rough Membership Functions. In: Advances in the Dempster-Shafer Theory of Evidence, pp. 251–271. John Wiley and Sons, New York (1994)
2. Stepaniuk, J.: Rough – Granular Computing in Knowledge Discovery and Data Mining. SCI, vol. 152. Springer, Heidelberg (2009)
3. Hassanien, A.E., Abraham, A., Peters, J.F., Kacprzyk, J.: Rough Sets in Medical Imaging: Foundations and Trends. In: Computational Intelligence in Medical Imaging: Techniques and Applications, pp. 47–87. CRC Press, USA (2008)
4. Michalski, R.: A Theory and Methodology of Inductive Learning. Artificial Intelligence 20(2), 111–161 (1983)
5. Wang, H., Zhou, M., William, Z.: A New Approach to Establish Variable Consistency Dominance—Based Rough Sets Based on Dominance Matrices. In: International Conference on Intelligent System Design and Engineering Application, Sanya, Hainan, pp. 48–51 (2012)
6. Ren, Y., Xing, T., Quan, Q., Chen, X.: Attributes Knowledge Reduction and Evaluation Decision of Logistics Centre Location Based on Rough Sets. In: 4th International Conference on Intelligent Computation Technology and Automation, Shenzhen, pp. 67–70 (2011)
7. Zaras, K., Marin, J.C., Boudreau-Trude, B.: Dominance-Based Rough Set Approach in Selection of Portfolio of Sustainable Development Projects. American Journal of Operations Research 2(4), 502–508 (2012)
8. Ke, G., Mingwu, L., Yong, F., Xia, Z.: A Hybrid Model of Rough Sets and Shannon Entropy for Building a Foreign Trade Forecasting System. In: 4th International Joint Conference on Computational Sciences and Optimization, Yunnan, pp. 7–11 (2011)
9. Lai, C.J., Wen, K.L.: Application of Rough Set Approach to Credit Screening Evaluation. Journal of Quantitative Management 12(1), 69–78 (2005)
10. Chao, D., Sulin, P.: The BSC Alarm Management System Based on Rough Set Theory in Mobile Communication. In: 7th International Conference on Computational Intelligence and Security, Hainan, pp. 1557–1561 (2011)
11. Hossam, A.N.: A Probabilistic Rough Set Approach to Rule Discovery. International Journal of Advanced Science and Technology 30, 25–34 (2011)
12. Qu, Z., Wang, X.: Application of Clustering Algorithm and Rough Set in Distance Education. In: 1st International Workshop on Education Technology and Computer Science, Wuhan, pp. 489–493 (2009)
13. Sheu, T., Chen, T., Tsai, C., Tzeng, J., Deng, C., Nagai, M.: Analysis of Students' Misconception Based on Rough Set Theory. Journal of Intelligent Learning Systems and Applications 5(2), 67–83 (2013)
14. Romero, C., Zafra, A., Luna, J.M., Ventura, S.: Association rule mining using genetic programming to provide feedback to instructors from multiple-choice quiz data. Expert Systems 30(2), 162–172 (2013)
15. Cole, J.: Using Moodle. O'Reilly (2005)