Web Usage Mining for Improving Students Performance in Learning Management Systems

Amelia Zafra and Sebastián Ventura

Department of Computer Science and Numerical Analysis,
University of Cordoba, Spain
{azafra,sventura}@uco.es

Abstract. An innovative technique based on multi-objective grammar guided genetic programming (MOG3P-MI) is proposed to detect the most relevant activities that a student needs to pass a course based on features extracted from logged data in an education web-based system. A more flexible representation of the available information based on multiple instance learning is used to prevent the appearance of a great number of missing values. Experimental results with the most relevant proposals in multiple instance learning in recent years demonstrate that MOG3P-MI successfully improves accuracy by finding a balance between specificity and sensitivity values. Moreover, simple and clear classification rules which are markedly useful to identify the number, type and time of activities that a student should do within the web system to pass a course are provided by our proposal.

Keywords: Web Usage Mining, Educational Data Mining, Multiple Instance Learning, Multi-objective Evolutionary Algorithm, Grammar Guided Genetic Programming.

1 Introduction

As an increasingly powerful, interactive, and dynamic medium for delivering information, the World Wide Web in combination with information technology has found many applications. One popular application has been for educational use, as in Web-based, distance or distributed learning. The use of the Web as an educational tool has provided learners and educators with a wider range of new and interesting learning experiences and teaching environments, that were not possible in traditional education. These platforms contain a considerable amount of e-learning materials and provide some degree of logging to monitor the progress of learning keeping track of learners' activities including content viewed, time spent at a particular subject and activities done. This monitoring trawl provides appropriate data for many different contexts in universities, like providing assistance for a student at the appropriate level, aiding the student's learning process, allocating relevant resources, identifying exceptional students for scholarships and weak students who are likely to fail.

A normal situation in the web-based learning systems is that instructors provide different resources and learners are encouraged to participate in the variety

N. García-Pedrajas et al. (Eds.): IEA/AIE 2010, Part III, LNAI 6098, pp. 439–449, 2010.

of activities. However, it becomes difficult for the instructors to track and identify all the activities performed by the learners and subsequently, it is hard to evaluate the structure of the course content and its effectiveness in the learning process. It would be very helpful to have an automatic tool for detecting those activities which improve the learning process and appraise the web-based course structure effectiveness. With this purpose, our research proposes a new method based on multiple instance learning (MIL) representation to classify students into two groups: the *low performance* students, who have a high probability of failing a course and the *high-performance* students, who have a high probability of passing a course. Our technique generates rules IF-THEN that allow the establishment of relationships between the student's work in web systems and passing a certain course. In this manner, instructors and students could be guided and recommended about what activities or resources would support and improve their learning.

The paper is structured as follows: section 2 reviews the works of data mining applied over academic scenarios; section 3 describes the problem that we aspire to solve showing its multiple instance representation and the available information to carry out this study; section 4 presents and explains the proposed model for solving the problem; section 5 shows experimental results and the rules obtained; finally, in section 6, the conclusion and further works are described.

2 Web Usage Mining in Academic Computing

Web usage mining [1] refers to non-trivial extraction of potentially useful patterns and trends from large web access logs. In the specific context of web-based learning environments, the increasing proliferation of web-based educational systems and the huge amount of information that has been made available has generated a considerable scientific activity in this field. Delavaire et al. [2], given the large amounts of data collected in academic institutions, propose a model with different types of education-related questions and the data-mining techniques appropriate for them. An example of a specific case study of clustering students who have similar characteristics (such as *self starters* and *high interaction*) is given in [3]. Anjewierden et al. [4] investigate the application of data mining methods to provide learners with real-time adaptive feedback on the nature and patterns of their on-line communication while learning collaboratively. Lazcorreta et al. [5] propose a new method for automatic personalized recommendation based on the behavior of a single user in accordance with all other users in web-based information systems. Chanchary et al. [6] analyze students' logs of a learning management system with data mining and statistical tools to find relationships between students' access behavior and overall performances. Finally, studies about the prediction of students' marks or their academic success have also been developed in this area [7,8,9,10]. Further information can be found in the survey carried out by Romero and Ventura [11].

3 Determination of Factors Influencing the Students' Academic Achievements

In this section, we will first present the definition of the problem that we want to solve. Then, we will describe the representation of the problem using MIL and finally, the specific data considered in this study will be set out in detail.

3.1 Definition of the Problem

A student could do different activities in a course to acquire and strengthen the concepts acquired in class. Later, at the end of the course, there is a final exam. A student with a mark over a fixed threshold passes a module, while a student with a mark lower than that threshold fails that module. With this premise, the problem consists of studying the work carried out by students to find interesting relationships that can suggest activities and resources for students and educators that can favour and improve both their learning and effective learning process.

To carry out this study the students will be classified into two groups: *high performance* students who have a high probability of succeeding and *low performance* students, with a high probability of failing (or dropping out). We will have information on every student which describes her/his work done on the web-based learning system. The idea is to determine the activities that are needed so that a student passes or fails the module taking into consideration the time dedicated and the number and type of activities done by the student during the course. Concretely, the types of activities considered in this study are quizzes considering the number of quizzes consulted, passed and failed, assignments considering the number of practices/tasks submitted or consulted and forums considering the number of messages sent and read.

3.2 A Multi-instance Perspective

In this problem, the different resources available to the student are not compulsory but rather are presented as a support to strengthen concepts before the final exam. Thus, each student can execute a different number of activities: a hard-working student may do all the activities available and on the other hand, there can be students who have not done any activities at all. Moreover, there are some courses which contain only a few activities and others which contain an enormous variety and number of them.

The characteristics of this problem make the information of each pattern different depending on the student and course. Traditional supervised learning would use the same structure with the same number of attributes for all patterns independently of the real information about each student and course and therefore, most patterns would have empty values. On the other hand, MIL [12] allows a representation that adapts itself perfectly to the concrete information available for each student and course, eliminating the missing values that appear in traditional representation. The key factor in MIL representation lies in the concepts

Table 1. Attributes for MIL representation of the student

INFORMATION ABOUT BAGS	
Attribute	*Description*
User-Id	Student Identifier.
Course	Course identifier.
FinalMark	Final mark obtained by the student in this course.
INFORMATION ABOUT INSTANCES	
Attribute	*Description*
Type Activity	Type of activity which represents the instance. Eight type of activities are considered: FORUMS: read, written or consulted, QUIZZES: passed or failed and ASSIGNMENTS: submitted or consulted.
TimeActivity	Time spent to complete the tasks of this type of activity.
NumberActivity	Number of activities of this type completed by the student.

of pattern and instances. In this learning a pattern (called bag) could contain different numbers of instances. Thus, the correspondence is one-to-several and not one-to-one as in traditional supervised learning. In this problem, each pattern or bag represents a student registered in a course. Each student is regarded as a bag which represents the work carried out. Each bag is composed of one or several instances and each instance represents the different types of work that the student has done. Therefore, each pattern will have as many instances as different types of activities that have been done by the student. Information about the attributes of bags and instances could be consulted in Table 1.

3.3 Case Study

This study employs the students' usage data from the web-based learning environment at Cordoba University considering Moodle platform [13]. The available information on the web usage log files for students was collected during an academic year from September to June, just before the Final Examinations. Seven different courses are considered with 419 students. The details about the different courses are given in Table 2, only considering about each student his/her identifier, activity with respect to forums, assignments and quizzes and final mark obtained in this course although more information is stored by the system.

4 The Proposed Model for Using Web Usage Mining in Education Data

In this section, we propose a new model based on multi-objective grammar guided genetic programming to predict the students' academic performance

Table 2. General Information about Data Sets

Course Identifier	ICT-29	ICT-46	ICT-88	ICT-94	ICT-110	ICT-111	ICT-218
Number of Students	118	9	72	66	62	13	79
Number of Assignments	11	0	12	2	7	19	4
Number of Forums	2	3	2	3	9	4	5
Number of Quizzes	0	6	0	31	12	0	30

based on features extracted from logged data in a web-based education system. The motivation to include this algorithm called MOG3P-MI is on one hand, the use of multi-objective proposal that allows us to find a set of optimal solutions considering different contradictory measurements. We search to find a solution with the best balance between sensitivity and specificity values to achieve the most accuracy models. The objective is to determine the most relevant resources to improve the learning process from the classification model obtained by this proposal. Thus, it is very important for the model to be precise and correctly classify both students who pass and those who fail the course, because otherwise the conclusions would not be significant. On the other hand, this algorithm allows to work in MIL scenario using a more flexible representation that eliminates the missing values of other representations which hinder algorithms from achieving the highest accuracy in the classification. Finally, another advantage of this proposal is that grammar-guided genetic programming (G3P) [14] is considered a robust tool for classification in noisy and complex domains and allows us to obtain representative information about the knowledge discovered to identify the most relevant activities for supporting and improving students' learning.

The design of the system will be examined in more detail in continuation with respect to the following aspects: individual representation, genetic operators, fitness function and evolutionary process.

4.1 Individual Representation

Individuals in our system represent classifiers which classify students into two classes (students who have a high probability of passing a course and students who have a low probability of doing so). The classifier is composed of IF-THEN rules whose antecedent determines the time, number and type of activities that the students have to do, if they want to have a high probability of passing the course. Those students who do not satisfy these requirements will have a low probability of passing the course. The rule generated has the following structure:

```
If ($condition$(student)) then
    the student will have a high probability of passing the course.
End-If
```

⟨S⟩ → ⟨condition⟩

⟨condition⟩ → ⟨cmp⟩ | **OR** ⟨cmp⟩ ⟨condition⟩ | **AND** ⟨cmp⟩ ⟨condition⟩

⟨cmp⟩ → ⟨op-num⟩ ⟨variable⟩⟨value⟩ | ⟨op-int⟩ ⟨variable⟩⟨value⟩⟨value⟩

⟨op-num⟩ → **GE** | **LT**

⟨op-int⟩ → **IN** | **OUT**

⟨variable⟩ → **TypeOfActivity** | **NumberOfActivity** | **TimeOfActivity**

⟨value⟩ → *Any valid value*

Fig. 1. Grammar used for representing individuals' genotypes in G3P-MI

The *condition* clause is represented by means of the grammar shown in Figure 1. Thus, the *condition* consists of several comparisons of attribute-value attached by conjunction and/or disjunction that inform about the number, type and time that should be dedicated by a student to be successful in a course.

4.2 Genetic Operators and Fitness Function

MOG3P-MI uses a crossover and mutator operator to generate new individuals in a given generation of the evolutionary algorithm based on selective crossover and mutation as proposed by Whigham [14].

The fitness function combines two commonly used indicators, namely sensitivity (Se) and specificity (Sp) to measure the classifier's effectiveness. Sensitivity is the proportion of cases correctly identified as meeting a certain condition and specificity is the proportion of cases correctly identified as not meeting a certain condition.

$$sensitivity = \frac{t_p}{t_p + f_n}, \quad specificity = \frac{t_n}{t_n + f_p}$$

where, t_p is the number of positive bags correctly identified, f_p is the number of positive bags not correctly identified, t_n is the number of negative bags correctly identified and f_n is the number of negative bags not correctly identified.

The goal of MOG3P-MI is to maximize both Sensitivity and Specificity at the same time. These two measurements evaluate different and conflicting characteristics in the classification process where a value of 1 in both measurements represents perfect classification.

4.3 Evolutionary Algorithm

The main steps of this algorithm are based on the well-known Non-Dominated Sorting Genetic Algorithm (NSGA2) [15] philosophy, but it is designed to use a grammar guided genetic programming and multi-instance learning to add more flexibility and clarity to the representation of the individuals. The general outline of our algorithm is shown in Algorithm 1.

Algorithm 1. MOG3P-MI Algorithm

```
Set $P_0$ = an initial population of rules, $A_0 = \phi$ (empty external set) and t = 0.
$P_0$ is sorted based on the concept of non-domination fronts.
Assign fitness to $P_0$, the fitness is equal to its non-domination level.
repeat
  $R_t = P_t \cup Q_t$. Combine parent and children population.
  All non-dominated fronts of $R_t$ are constructed.
  repeat
    if $|P_{t+1}| < N$ then
      Calculate crowding distance of in Front $i$, $F_i$.
      $P_{t+1} = P_{t+1} \cup F_i$
    end if
  until $|P_{t+1}| < N$
  Sort in descending order using the crowding distance value.
  $P_{t+1} = P_{t+1}$ $[0 : N]$. Choose the N first elements of $P_{t+1}$.
  Use selection, crossover and mutation to create a new population, $Q_{t+1}$.
  Set t = t + 1
until acceptable classification rule is found or the specified maximum number of
  generations is reached.
```

5 Experimentation and Results

The commented case study is used to evaluate our proposal and to compare it with other similar models. Our primary objective is to determine whether the results of our model are accurate enough compared to the rest of the proposals. Then we determine the most relevant activities to help the professor and students to improve the learning process by analyzing the information provided by our system about the number, type and time that student has to devote to have a high probability of passing a course.

5.1 Results and Discussion

The most relevant proposals based on MIL presented to date are considered to solve this problem and compared to our proposal designed in JCLEC framework [16]. The paradigms compared include methods based on diverse density, on Logistic Regression, on support vector machines, on distance, on rules, on decision trees, on naive Bayes and on evolutionary algorithms [17]. More information about the algorithms considered could be consulted at the WEKA workbench [18] where these techniques are designed.

All experiments are carried out using 10-fold stratified cross validation and the average values of accuracy, sensitivity and specificity [19] are reported. Moreover, evolutionary algorithms performed five different runs with different seeds and the average values of sensitivity, specificity and accuracy are reported. Table 3 shows the average results of accuracy, sensitivity and specificity values for each method considered in the study.

Table 3. Experimental results to compare MOG3P-MI with other methods based on MIL

Paradigm based on	*Algorithm*	*Accuracy*	*Sensitivity*	*Specificity*
***R**ules*	PART	0.7357	0.8387	0.5920
	AdaBoostM1&PART	0.7262	0.8187	0.5992
	Bagging&PART	0.7167	0.7733	0.6361
	AdaBoostM1&PART	0.7071	0.7735	0.6136
	PART	0.7024	0.7857	0.5842
***N**aive Bayes*	NaiveBayes	0.6786	0.8515	0.4371
***S**VM*	SMO	0.6810	0.8644	0.4270
***D**istances*	MIOptimalBall	0.7071	0.7218	0.6877
	CitationKNN	0.7000	0.7977	0.5631
***D**ecision Tree*	DecisionStump	0.6762	0.7820	0.5277
	RepTree	0.6595	0.7127	0.5866
***L**ogistic Regression*	MILR	0.6952	0.8183	0.5218
***D**iverse Density*	MIDD	0.6976	0.8552	0.4783
	MIEMDD	0.6762	0.8549	0.4250
	MDD	0.6571	0.7864	0.4757
***E**volutionary Algorithms*	G3P-MI	0.7429	0.7020	0,7750
	MOG3P-MI	**0.7952**	**0.7209**	**0.8500**

As can be seen, MOG3P-MI obtains the most accurate models. This classification problem has an added difficulty since it deals with a variety of courses with different numbers and types of exercises which makes it more costly to establish general relationships among them. Nonetheless, MOG3P-MI in this sense is the one that obtains the best trade-off between the two measurements, obtaining the highest values for specificity without a relevant fall in sensitivity values. If we observe the results of the different paradigms, it can be seen how they optimize the sensibility measurement in general at the cost of a decrease in the specificity values (excepting G3P-MI that also achieves a balance between the different measurements, although the results are worse). This leads them to an incorrect prediction of which students will not pass the course.

Moreover, MOG3P-MI obtains interpretable rules to find pertinent relationships that could determine if certain activities influence the student's ability to pass, if spending a certain amount of time on the platform is found to be an important contributing factor or if there is any other interesting link between the work of the student and the final results obtained. These relationships will be studied in the next section.

5.2 Determining Activities That Improve the Learning Process

One of the advantages of our system is that it generates a classifier consisting of IF-THEN rules which provide knowledge about the requirements that a student must satisfy to have a high probability of passing a course. These rules are simple, intuitive, easy to understand and provide representative information about the type, number and time necessary for a student to pass the course.

By the following examples, we are going to study the information provided by the model and analyze the activities that are the most efficient for acquiring knowledge.

IF [$((\mathit{TimeOfActivity} < 1508) \wedge (\mathit{NumberOfActivity} = 7)) \vee$
$(((\mathit{TypeOfActivity} = \text{QUIZ\_P}) \wedge (\mathit{NumberOfActivity} \geq 6)) \vee$
$(\mathit{TimeOfActivity} \in [3007, 9448])$]
THEN *The student has a high probability to pass the course.*
ELSE *The student has a low probability to pass the course.*

According to this rule, students who have a high probability of passing a course have to satisfy one of the following requirements: *a)* do at least six quizzes correctly, *b)* perform seven activities of some kind devoting at least 1508 minutes in the web-based system consulting resources or making activities or *c)* spend between 3007 and 9448 minutes to perform different activities in the web-based system or consult the resources available.

IF [$((\mathit{NumberOfActivity} \in [3, 12]) \wedge (\mathit{TypeOfActivity} = \text{QUIZ\_P}) \wedge$
$(\mathit{TimeOfActivity} \in [516, 1727])) \vee (\mathit{TimeOfActivity} \in [2998, 7143]) \vee$
$((\mathit{TypeOfActivity} = \text{QUIZ\_P}) \wedge (\mathit{NumberOfActivity} \geq 7))$]
THEN *The student has a high probability to pass the course.*
ELSE *The student has a low probability to pass the course.*

Another of the rules obtained indicates that to pass a course it is necessary to satisfy some of the following constraints: *a)* perform at least three quizzes correctly justifying a time on the web-based system between 516 and 1727 minutes, *b)* take at least seven quizzes, or *c)* have spent at least between 2998 and 7143 minutes on the web system carrying out different activities or consulting the resources.

IF [$((\mathit{NumberOfActivity} \geq 8) \wedge (\mathit{TypeOfActivity} = \text{QUIZ\_P})) \vee$
$(\mathit{TimeOfActivity} \in [3009, 6503])$]
THEN *The student has a high probability to pass the course.*
ELSE *The student has a low probability to pass the course.*

Finally, the last rule that we analyzed indicates that to pass a course it is necessary: *a)* perform at least eight quizzes correctly or *b* devote between 3009 and 6503 minutes to consult resources and make activities on the system.

We see that in general all the rules establish similar requirements. In conclusion, the development of quizzes is one of the most crucial activities in all rules to predict whether a student will pass a course, the number of them is a value approximating seven correct quizzes. Therefore, the relevance of performing quizzes correctly becomes obvious to consolidate the students' knowledge. In the case of quizzes that are not carried out because the course has no available quizzes or because students have carried out quizzes but not correctly, or because directly the student has not wanted to do them, it is necessary to justify at between least 3000 and 7000 minutes on the web-based system consulting different resources or doing different activities (with this amount of time would be indifferent the particular type or number of activities that student should do). Finally, if the time spent on the system is lower, it is necessary to specify a minimum number of activities carried out. This number is nearly to seven different activities during the time devoted to the system, although the type of activities do not matter (therefore they could be both assignments and quizzes and forums).

6 Conclusion

In this paper a new method to determine the most relevant activities to get a high probability of passing a course based on the students' work carried out on a web-based learning system is proposed. The proposal consists of multi-objective grammar guided genetic programming algorithm, MOG3P-MI, in a MIL scenario. The most representative algorithms in MIL are compared with our proposal and experimental results show that MOG3P-MI achieves the most accurate models finding a solution with a balance between sensitivity and specificity values. Moreover, representative information is provided that allows us to see which activities presented to students are really relevant for improving the learning process and allow students to consolidate the concepts studied in the classroom to then pass the course.

The results obtained are very interesting, but the work only considers if a student passes a course or not. It is would be interesting to expand the problem to predict students' grades (classified in different classes) in an e-learning system. Another interesting issue consists of determining how soon before the final exam a student's marks can be predicted. If we could predict a student's performance in advance, a feedback process could help to improve the learning process during the course.

Acknowledgments

The authors gratefully acknowledge the financial support provided by the Spanish department of Research under TIN2008-06681-C06-03 and P08-TIC-3720 Projects.

References

1. Srivastava, J., Cooley, R., Deshpande, M., Tan, P.N.: Web usage mining: discovery and applications of usage patterns from web data. SIGKDD Explor. Newsl. 1(2), 12–23 (2000)
2. Delavari, N., Beikzadeh, M., Shirazi, M.: A new model for using data mining in higher educational system. In: ITHET'04: Proceedings of 5th International Conference on Information Technology Based Higher Education and Training, Istanbul, Turkey (2004)

3. Luan, J., Zhao, C.M., Hayek, J.: Exploring a new frontier in higher education research: A case study analysis of using data mining techniques to create nsse institutional typology. California Association for Institutional Research, California, USA (2004)
4. Anjewierden, A., Kollöffel, B., Hulshof, C.: Towards educational data mining: Using data mining methods for automated chat analysis to understand and support inquiry learning processes. In: ADML'07: International Workshop on Applying Data Mining in e-Learning, pp. 27–36 (2007)
5. Lazcorreta, E., Botella, F., Fernández-Caballero, A.: Towards personalized recommendation by two-step modified apriori data mining algorithm. Expert Systems with Applications 35(3), 1422–1429 (2008)
6. Chanchary, F., Haque, I., Khalid, S.: Web usage mining to evaluate the transfer of learning in a web-based learning environment. In: WKDD'08: International Workshop on Knowledge Discovery and Data Mining, pp. 249–253 (2008)
7. Fausett, L., Elwasif, W.: Predicting performance from test scores using backpropagation and counterpropagation. In: WCCI 1994: IEEE World Congress on Computational Intelligence, Washington, USA, pp. 3398–3402 (1994)
8. Minaei-Bidgoli, B., Punch, W.: Using genetic algorithms for data mining optimization in an educational web-based system. Genetic and Evolutionary Computation 2, 2252–2263 (2003)
9. Kotsiantis, S., Pintelas, P.: Predicting students marks in hellenic open university. In: ICALT'05: The 5th International Conference on Advanced Learning Technologies, Kaohsiung, Taiwan, pp. 664–668 (2005)
10. Superby, J., Vandamme, J., Meskens, N.: Determination of factors influencing the achievement of the first-year university students using data mining methods. In: EDM 2006: Workshop on Educational Data Mining, Hong Kong, China, pp. 37–44 (2006)
11. Romero, C., Ventura, S.: Educational data mining: A survey from 1995 to 2005. Expert System Application 33(1), 135–146 (2007)
12. Dietterich, T.G., Lathrop, R.H., Lozano-Perez, T.: Solving the multiple instance problem with axis-parallel rectangles. Artifical Intelligence 89(1-2), 31–71 (1997)
13. Rice, W.H.: Moodle e-learning course development. Pack Publishing (2006)
14. Whigham, P.A.: Grammatically-based genetic programming. In: Proceedings of the Workshop on Genetic Programming: From Theory to Real-World Applications, Tahoe City, California, USA, September 1995, pp. 33–41 (1995)
15. Deb, K., Agrawal, S., Pratap, A., Meyarivan, T.: A fast elitist non-dominated sorting genetic algorithm for multi-objective optimisation: NSGA-II. In: Deb, K., Rudolph, G., Lutton, E., Merelo, J.J., Schoenauer, M., Schwefel, H.-P., Yao, X. (eds.) PPSN 2000. LNCS, vol. 1917, pp. 849–858. Springer, Heidelberg (2000)
16. Ventura, S., Romero, C., Zafra, A., Delgado, J.A., Hervás, C.: JCLEC: A java framework for evolutionary computation soft computing. Soft Computing 12(4), 381–392 (2007)
17. Zafra, A., Ventura, S.: G3P-MI: A genetic programming algorithm for multiple instance learning. Information Science (submitted 2009)
18. Witten, I., Frank, E.: Data Mining: Practical machine learning tools and techniques, P2nd edn. Morgan Kaufmann, San Francisco (2005)
19. Kantardzic, M.: Data Mining. Concepts, Models, Methods and Algorithms. John Wiley and Sons, Chichester (2003)