Educational additions to an Open Source Virtual Learning Environment (VLE) for the Greek Schools' Network

Dr. Avgoustos Tsinakos
Associate Professor, Dep. of Industrial Informatics
T.E.I. of Kavala Greece
tsinakos@teikav.edu.gr

John Papaioanou
MSc. University of Surrey UK
pj@moodle.org

Abstract

This study describes one of the most useful features that have been developed as part of the Greek Schools' Network (GSN) project, in order to enrich the educational profile of the Moodle environment. The feature, so called "Assessment Module",automatically generates individual student progress reports for any student, based on course data and a predefined student model, and provides tools for managing an annotated history of these reports.

1. Introduction

A number of pedagogical drawbacks have been reported in the literature regarding the scenario of Asynchronous Distance Education. Delay responses, hazy monitoring of the students' performance progress, lack of student modelling processes etc. are commonly referred as critical problems [6],[7],[8],[9],[10],[11],[12],[15]. In an effort to identify the most appropriate Open Source VLE for the Greek Schools' Network project [5] a research study was conducted [13]. A number of nine Open Source VLE (ILIAS 2.3, Cose 2.061, KEWL 1.1.0, Moodle 1.0.8.1, Fle3 1.4.1, Manhattan 2.1, Claroline 1.3.1, CoMentor 1.0, Eledge 1.8) have been compared towards eighty-one evaluation criteria that have been derived by a number of organizations, institutions and researchers such as, the Western Cooperative for Educational Telecommunications [14], the Centre for Curriculum Transfer and Design [1], the Higher Education and New Technologies Center [4], Centre for Flexible Learning [2].

Moodle platform [3] was selected as the most appropriate platform for the GSN Moodle for Asynchronous Distance Education services for more than 10.500 schools in Greece (an analysis of the justification reasons are out of the scope of the current paper). To remedy some of the above mentioned drawbacks of Asynchronous Distance Education, ten further developments have been decided to be implemented in order Moodle to reach the desired standards of the Greek Ministry of Education.

These developments were, Calendar Module, Course Participants Module, Course Wizard Module, Questions Module*, Private Messages Module, Document Management System Module, Assessment Module*, Educational Profile Module, Personal E-mail Module, Student Activity Module. Among these, the Assessment Module is further analyzed in the current paper.

2. Implementation Purpose of Assessment Module

The primary purpose of the Assessment Module is allowing teachers to have a summarized report that provides a clear and concise view of the students' progress and educational profile, without having to manually review a possibly large number of assessment criteria. Furthermore, since this report is automatically generated in natural language, it is capable of spotting patterns in the students' progress and work that may not be very obvious from a simple list of grades and assessments. Such patters may not be easily detectable by the teacher because of the large amount of data that need to be reviewed and the repetitive process of evaluating numeric results, a task at which humans are generally not very effective.

These two benefits were the basic motivation for developing the Assessment Module. However, we need to stress the importance that these benefits can have for the educational process within a Virtual Learning Environment. Despite its many advantages, a VLE is definitely lacking in some respects when compared to a regular teaching class. The main issue here is that it is difficult for the teacher in a VLE to build student profiles and detect interesting patterns in the students' behavior. This is reasonable since in e-learning, the teacher does not receive the physical clues and stimuli from the students, as is the case in traditional learning environments. Interesting patterns, when used in this context, refers to (among others) extraordinary performance, very poor performance, difficulty dealing with a specific subtopic within a module, exceptional diligence, cooperative spirit, etc. The recognition of such student's special characteristics is essential for a personalized and more fruitful educational process; the teacher can use this information to cope with the different problems the students may have or reward their merits and help boost their efforts. In this context, the Assessment Module greatly helps the teacher to track

1-4244-0212-3/06/$20.00/©2006 IEEE

down individual student learning patterns and reach appropriate educational conclusions about them. In addition, the facility of being able to manually edit the automatically generated report and maintain a history of these updates can help provide an even more concrete view of the student's profile and progress. The benefit that the VLE and the learning process enjoys from the Assessment Module is therefore increased.

3. Structure and components of the system

The automatically generated natural language report is based on the recorded data regarding the presence and achievements of each student in various activities of a module. Because strictly defining the exact character of evaluated information would require a different choice of words for each type of learning activity, the word "performance" is hereafter used to describe such student-course interaction results. This use of the word in this context should not be confused with its traditional meaning of "achievement level", because in many learning activities (e.g. forums) the concept of achievement simply does not apply.

The current implementation is based on criteria related to three broad kinds of student activities. These are drawn from the performance on all kinds of written assignments, the participation in forums, and the inquisitive queries submitted (via the Questions module). Nevertheless, the system has been built in such a way that it can be expandable and new criteria be relatively easily embedded in it. A discussion of the system's structure follows.

The system has two components, one is the inference component and the other is the natural language generation component. In this current implementation these are very much dependent on each other but in the future it is planned that the coupling between these two modules be loosened, and thus allowing the design to become more modular and organized.

4. The inference component

The inference component is a rule-based forward chaining system. A rule based system uses a working memory of facts, a knowledge base, and a set of rules associated with specific preconditions. If the preconditions for a rule are met, the rule activates and carries out some proper action, like adding new facts to the working memory. In a forward chaining rule-based system, rules can be activated (possibly repeatedly) depending on facts that other rules have published to the working memory. This activation could lead to more facts being added, which could allow more rules to activate, and so on. The process stops when no more rules can be activated. At that point, the final set of facts will contain a subset of conclusions, or inferred facts. Since the forward chaining system needs some initial facts to activate at least one rule, it is understood that the final inferred conclusions are a function of the initial facts (our data set, which consists of educational information) and the rule set applied to it (the student model).

Strictly speaking, in the current implementation we did not include rules that cannot be activated using only information from the initial data set. The rules that have been so far implemented deposit facts on the working memory which no other rule looks for, thus in its present form our system is not exactly a forward chaining system. However, its design allows the easy addition of rules that use facts from the working memory, which would transform it into a real forward chaining system. This task, which is quite straightforward due to the modularity of the implementation, will necessarily lead to more complex student models. Since the present student model has not been tested on large real-life datasets and refined, taking such steps right now might be considered premature.

5. The natural language generation component

The second component, the natural language generator, processes the facts inferred from the initial data set and generates the report that the teacher sees. Natural language generation is currently considered a very difficult problem, studied mainly by Human Computer Interaction and Artificial Intelligence experts. However, the set of words and expressions needed for a student progress report is limited in comparison to a generalized application scenario and thus our report generating system does not necessarily have to be that complex. The current implementation is based on some simple interpretation of the generated facts.

Each fact (conclusion) produces in the end a simple sentence of standard form. Sentences are associated with a small number of characteristics, like for example their positive, negative, or indifferent tone. This enables the natural language generator to decide how to run these sentences together in such a way as to maintain coherence and human-like flow of language. The generator keeps track of an internal state which attempts to describe in a simple way the current tone of the generated report, and subsequent sentences both get modified according to and in turn themselves modify this state. Suitable expressions are randomly drawn upon from a predefined pool built into the system for each case where two sentences need to be run together. Despite the simplicity of the design, the final result is that the system can create quite good natural language texts that reflect in a quite compact way the results generated by the inference component.

6. The set of rules

The efficiency of the system depends critically on the inference component, which in turn depends on the rule selection strategy and the set of rules (i.e., the student model). One has to be very careful when designing a rule based inference system like the one described in this paper, since a bad selection of rules and inference strategy may introduce severe problems, usually of computational character and related with overflow of results. We will not consider this problem here, and instead focus on the subject of useful conclusion detection, i.e. the issue of finding appropriate rules to draw useful results from our data set. The most important factor for getting such results is the strategy for designing suitable rules for the system. The formation of the rules is clearly a question of what kind of behavior the teacher would like to detect and what characteristics of the student's profile he wants to concentrate on.

In the current implementation, we have arbitrarily created some rules based on intuition. We consider that this is convenient for the time being but clearly the subject needs to be further discussed. We will now examine some of the rules that our system uses and we will come back to this issue after highlighting some interesting points.

As mentioned earlier, the current implementation of the inference system uses criteria from four different kinds of activities: assignments, forums, questions and quizzes. A set of rules is assigned to each of these activities. For example we have the following simple set of rules related with the performance of the student in assignments:

The first rule classifies the average grade of the student in one of four bands combined with a statement about the certainty of the result. That is, if the average grade is above 75% the result is "very well", if the average grade is between 60% and 75% the result is "quite well", if the average grade is between 40% and 60% the result is "satisfactorily" and if the average grade is below 40% the result is "poorly". Also, if the number of graded assignments is bigger that two and there is no assignment with more than a 10% deviation from the mean, the student's performance is characterized as "consistent". Relating these with the aforementioned form of the rules, we can say that the conditions are the average grade, the number of graded assignments and the divergence of the individual grades from the mean. The resulting facts that are added to the list of facts are the classification of the average as "very well", "quite well", "satisfactorily" and "poorly" and the characterization of the result as "consistent" or not.

The second rule detects assignments where the student has performed very poorly compared to the others. The rule first checks if there are enough graded assignments so that any conclusions reached could be considered "safe" (a simple check for the presence of 3 or more graded assignments) and then sees if any of the grades drops below 50% of the median. Since the median of a numerical set is generally not affected by the presence of unusually high or low numbers as is the case for the mean, this strategy can detect their presence.

The third rule checks the fluctuation of the student's grades. It first classifies each grade in one of 3 bands. It then checks if subsequent grades are classified on different bands. If the grades of two subsequent assignments are classified on the same band, there is no fact added to the working fact set. If the grades are classified in neighboring bands, i.e. the first and the second, or the second and the third, a fact that merely mentions this is added to the working fact set. Finally, if the grades are classified in bands that are not neighboring, i.e. the first and third, a fact that declares a strong comment is added on the set. Obviously, this system can be easily refined with more than three bands and/or other improvements. Furthermore, when there is a performance difference, the topic of the assignment is taken into account, because it can conceivably help us reach a better conclusion: if the two assignments are on the same topic, it is inferred that the student has probably not worked enough for one of them., whereas if the topic is different, it is inferred that the difference is due to progress problems (i.e. the student does not yet have full grasp of the new topic).

These rules as a whole do not aspire to be distinguished for their sophistication, and the implemented rules for the other types of activities examined by the inference component are also similar in orientation and level of complexity. At present, they can all be considered as very simple feature detectors, since they look for certain characteristics in the performance and the achievements of students. One may argue that these are too simple and thus not very suitable for the inference of important educational conclusions. It needs to be clear that we understand that a much better set of rules that produce more beneficial educational results can be used. However, due to the rich ways of interaction between rules that a forward-chaining system allows, the problem of finding a correct, or even just a better, rule set is far from trivial. For example, when adding a new rule to the system, one has to consider the possible interactions of the new rule with those already present, and also evaluate the possible "transfer of decision-making power" from one rule to another. Given that any interaction between two rules can conceivably be performed by utilizing an arbitrary number of intermediate results generated from any other rule, it follows that this process can be whole research topic by itself. At this time, our decision was to focus on the construction of a good, robust and modular inference system from a software engineering point of view, and not prematurely try to optimize the student model. With such a system available to work on, the addition of new rules and the enhancement of the decision-making logic is pretty straightforward technically; thus the way is open for innovation and research on the topic of the student model.

7. Evaluation and future work

In its present state the Assessment Module is a very useful feature of Moodle. Its merits were mentioned in the first part of the paper. Nevertheless, it can be further enhanced and indeed it was designed with future enhancement in mind. As we have already said, the first step could be to build a more useful set of rules that use the forward chaining capacity of the inference system, making the Assessment Module capable of reaching much more useful conclusions.

In light of the above, a big step in the development of the system would be to get the rules from an external source. One option would be to conduct a series of interviews with teachers and educational experts and elicit from them a possibly suitable set of rules that produces interesting results. Another option would be to create the rules from a collection of data. This means that, given a large amount of student behavior data and assign properties to the various records we can use a rule discovery tool like some kind of a Decision Tree or the APRIORI algorithm to automatically create rules that detect the kind of behavior that we are interested in. In this case, the difficult part would be to collect a set of records describing various aspects of student behavior and appropriate labels for these records that assign properties to and characterize them (these could be provided from a teacher at an initial stage). After retrieving the automatically generated rules, we will be able to use them on data collected from new students and make inferences about them as well. Technically, this is a supervised approach since we assume that except for the records that describe the student behavior we also have some labels that characterize each record.

8. References

[1] Centre for Curriculum, Transfer and Technology (C2T2) and Western Cooperative for Educational Telecommunications .Edutools: Web-based Resource for the Higher Education Community, 2003, Available online at http://www.edutools.info/index.jsp.

[2] CFL Integrated Web-based Course Delivery Tools, Center for Flexible Learning, 2003 Available online at http://www.cfl.mq.edu.au/.

[3] Dougiamas, M. and Taylor, P. Moodle: Using Learning Communities to Create an Open Source Course Management System, EDMEDIA 2003. Available online at http://dougiamas.com/writing/edmedia2003/

[4] Edutech, Higher Education and New Technologies, Evaluation of Learning Management Systems, 2002. Available online at http://www.edutech.ch

[5] Greek Ministry of Education. Greek School Network Description, 2005. Available online at http://www.sch.gr/en/

[6] Hall, B. Web-based training cookbook. New York: Wiley Computer Publishing 1997.

[7] Harasim, L. Online education: An environment for collaboration and interllecual amplification. In L. Jarasim (Ed.), Online education: Perspectives on a new environment. New York:Praeger, 1990, p. 39-64.

[8] Hazari S. Evaluation and Selection of Web Course Management Tools, Robert H. Smith School of Business, University of Maryland, 1998. Available online at http://sunil.umd.edu/webct/default.html.

[9] Khan, B.H. Ed.. Web-based instruction. Englewood Cliffs, New Jersey: Educational Technology Publications, Inc., 1997,p. 41-46.

[10] Pernici, B. and Casati, F. The design of distance education applications based on the World Wide Web, in B. H., 1997.

[11] Pritichard, C. L. The ASTD handbook of instructional technology. New York: MacGraw-Hill, Inc., 1998

[12] Relan, A. and Gillani,B. B. Web-based instruction and the traditional classroom: Similarities and differences, in B. H., 1997.

[13] Tsinakos A. The puzzle of Virtual Learning Environments: what criteria should be present in the ideal VLE?" , Turkish Online Journal of Distance Education-TOJDE, ISSN 1302-6488, 2004, 5 (2). Available online at:
http://tojde.anadolu.edu.tr/tojde14/articles/tsinakos.htm

[14] WCET Western Cooperative for Educational Telecommunications, 2003. Available online at: http://www.wcet.info/ projects/

[15] Wulf, K. Training via Internet: Where are we? Training & Development, 1996, 50 (5), 50-55.