A Framework for Constructing Adaptive Web-Based Educational Systems

Marek Obitko
Dept. of Cybernetics, FEE, Czech Technical University, Prague, Czech Republic
obitko@labe.felk.cvut.cz

Lubomír Kurz
Neuralgen s.r.o., Brno, Czech Republic
http://www.neuralgen.com/
lubomir@kurz.cz

Igor Glűcksmann
Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic
igluck@atlas.cz

Abstract

Adaptive educational systems are often built from scratch with no general methodology that would enable the system to be easily reused in another domain [5]. We propose a general methodology to structure the taught domain into so-called "knowledge units" that can be treated in a unified way. Individualization of the learning process for a particular student consists of two layers - a cognitive and a presentation layer.

We will briefly describe our framework for the construction of web-based educational systems. A course in elementary mathematical functions is presented as an example of our approach.

1. Goals of Learning

In a typical educational learning process, a student is expected to acquire knowledge and, more importantly, to apply the new knowledge to problem solving.

Since we cannot measure the student's knowledge directly, we have to build a student model by observing outputs (i.e. answers) resulting from inputs (i.e. questions) in the learning process. The goal of learning is to make the student's answers correspond with the required outputs. A flexible question-answer mode, which individualizes the presented information based on feedback from the student, can be employed here very efficiently.

To summarize, learning can be understood as the adaptation of the student's ability to react correctly to presented tasks. This adaptation can be reached and measured by a controlled dialogue with feedback.

2. Learning System Knowledge Base

The most visible result of the learning process is how correct the student's answers are. Based on this, the setting up of an appropriate question-answer structure is a basic part of defining the knowledge base. In fact, a knowledge base can always be represented by a question-answer set and any knowledge item can be presented in a question-answer mode. We can structure the whole course in units built around the questions expected to be answered correctly as a result of learning.

2.1. Knowledge Units

In our view, a knowledge unit is a compact set of related knowledge items that can be taught and examined independently. Knowledge units can be defined with various granularity (e.g. one unit can correspond to one paragraph or to one chapter in a book) for different groups of students. The granularity of a knowledge unit can be changed dynamically.

Considerable attention has been paid to various relations between the knowledge units, which can result in quite a complicated structure of the knowledge domain [5]. The actual effect of such structures is often not worth the effort. In our approach, we use just one type of relation, prerequisite or predecessor.

2.2 Knowledge Unit Functions

A knowledge unit teaches the embedded knowledge by presenting to the student various "screens", to which the student can react. The simplest unit could use only a question and an answer while a more complicated unit may contain a deeper explanation, hints, examples etc. at several levels (see below) and/or a generator of sample questions (exercises) with corresponding answers.

A unit presents embedded knowledge to the user, registers his reactions, updates the mastery level, and produces another presentation. When the required mastery level is reached, the unit transfers control to another unit.

3. Individualization for a Particular Student

In order to achieve maximal efficiency in a learning process, each student needs his own personalized treatment. It has been shown (e.g. [2]) that we cannot expect the student to search proactively for the right information needed to answer questions from an unknown domain. Moreover, the ability to take in the information presented to the student depends on his or her background. It means there is no general presentation pattern meeting the needs of every student.

Proceedings of the IEEE International Conference on Advanced Learning Techniques (ICALT'01)
0-7695-1013-2/01 $10.00 © 2001 **IEEE**

The individualization for a particular student can be obtained by a combination of changes at the cognitive and the presentation layer.

3.1. Cognitive Individualization

The cognitive state indicates how the student has mastered the domain [4]. The cognitive student model is expressed by the set of student's mastery levels of knowledge units. This model is used to control the flow between units. A possible algorithm for cognitive individualization follows:

1. Select the unit that has not been mastered yet and that has all prerequisite units mastered.
2. Let the unit deliver a presentation. Evaluate the user's answer/reaction and update the cognitive student model (note that the student's answer may also influence the mastery level of other units).
3. Repeat from step 1 until all units have been mastered.

Such cognitive individualization together with presentation individualization enables us to repeat the presentation at different levels and in different styles (as discussed below) until the required mastery level is achieved. It also enables students to learn at their own pace without any unnecessary presentations.

3.2. Presentation Individualization

Every item of knowledge can be presented in several ways with differing levels of comprehension, i.e. the complexity and amount of textual and other explanation, and it can also be presented in various forms. We assume that the expected comprehensibility can be parameterized and all possible presentations generated by the unit can be ordered by these parameters. Ordered presentations create presentation levels; each knowledge unit can have several levels. The goal is to lead each student through the levels that are most appropriate for the individual.

4. Case Study - Elementary Mathematical Functions

Using the methodology described above, we have created several particular teaching systems implemented in the WWW environment (reasons for this environment are summarized e.g. in [3]). In this section, we will describe a case study that uses our domain-independent Java framework.

The core of the framework is an invisible Java applet that generates presentations for the user as dynamic HTML pages. These pages use their own embedded applets from [1,6] for displaying graphical information.

The goal of the course is to teach the students to recognize elementary mathematical functions, especially relations between a formula and the corresponding graph. Each knowledge unit corresponds to one type of function. For each function, the student is expected to know its name, general formula, graph, and to recognize the values of the parameters of the function from the graph and vice-versa. Both knowledge units and their presentation levels are placed in order of expected difficulty to enable cognitive and presentation individualization as described above.

This system will be used at the Mendel University in Brno to help first-year students prepare for a part of their Mathematics I exam.

5. Concluding Remarks

We have presented a methodology for constructing an intelligent adaptive tutoring system together with a fully functional, flexible, web-based framework based on this methodology. This framework was tested on several domains. One domain, an elementary functions course, was described briefly. The advantage of this methodology is the simplicity in authoring the system while keeping to the main goal of the teaching process - to teach the student to use the acquired knowledge on practical examples. We do not force authors to develop unreal, complicated structures of the system control; the system uses data collected from students' responses to adapt a learning sequence to particular student's needs. Currently we are planning a careful evaluation of the system based on real data obtained from students.

Acknowledgements

This research was supported by Neuralgen s. r. o., Czech Republic. The first author was supported by the MSMT Grant No. 212300013.

References

[1] "HotEqn – The IMGless Equation Viewer", http://www.esr.ruhr-uni-bochum.de/VCLab/software/HotEqn/HotEqn.html

[2] Vincent Aleven and Kenneth Koedinger. Limitations of Student Control: Do Students Know When they Need Help?, In [5], pages 292-303

[3] Peter Brusilovsky. Adaptive Hypermedia: From Intelligent Tutoring Systems to Web-Based Education. In [6], pages 1-7.

[4] Nicola Capuano, Marco Marsella and Saverio Salerno. "ABITS: An Agent Based Intelligent Tutoring System for Distance Learning", in C. Peylo (ed.), *Proceedings of the International Workshop on Adaptive and Intelligent Web-based Educational Systems*, Montréal, Canada. Pages 17-28.

[5] Gilles Gauthier, Claude Frasson and Kurs VanLehn (Eds.): *Intelligent Tutoring Systems, ITS2000*, Montréal, 2000. Springer-Verlag Lecture Notes in Computer Science Vol. 1839.

[6] Marek Obitko and Pavel Slavík. Visualization of Genetic Algorithms in a Learning Environment. In J. Žára (ed.), *Spring Conference on Computer Graphics, SCCG'99*, pages 101-106, Bratislava, 1999. Comenius University.