Adaptive courseware model for intelligent e-learning systems

Ani Grubišić
Computer Science Department
Faculty of Science, University of Split, Croatia
ani.grubisic@pmfst.hr

Slavomir Stankov
Computer Science Department
Faculty of Science, University of Split, Croatia
slavomir.stankov@pmfst.hr

Branko Žitko
Computer Science Department
Faculty of Science, University of Split, Croatia
branko.zitko@pmfst.hr

Abstract *:*

This paper describes an Adaptive Courseware Tutor – an intelligent tutoring system based on stereotypes, Bayesian networks and Bloom's knowledge taxonomy. The main feature of our approach is the automatization of learning object generation and courseware adaptivity in every stage of learning and teaching process. The student module is enhanced by double stereotypes based on student's knowledge level and on Bloom's knowledge taxonomy, as well as, by Bayesian networks. The tutor module is responsible for the automatic generation of courseware elements, their dynamic selection and sorting, as well as their adaptive presentation using templates for statements and questions. In order to evaluate the model's effectiveness, a controlled experiment with a large sample was conducted.

Keywords-component; *Intelligent tutoring systems, adaptive e-learning systems, adaptive courseware, stereotypes, Bloom's knowledge taxonomy*

I. INTRODUCTION

Since one of the main features of Intelligent Tutoring Systems (ITS) [1] is adaptivity and since e-learning systems generally do not behave adaptively, we have decided to realize adaptivity of educational content in e-learning systems in order to get them as close as possible to ultimate goal of tutoring. The main feature of our approach is the automatization of courseware adaptivity in every stage of learning and teaching process. Adaptive systems [2] and adaptive instruction [3] have influenced the development of adaptive educational systems (AES) or adaptive e-learning systems. These systems adapt the process of learning, teaching and testing knowledge to different characteristics of a student. The intelligent tutoring systems (ITS) and adaptive educational hypermedia systems (AEHS) are two best-known representatives of adaptive e-learning systems [4]. The intelligent tutoring systems have had all necessary characteristics to lead in the development of adaptive e-learning systems. However, their inflexibility and development cost prevent this from happening [5].

This paper presents a new approach to realization of adaptivity in the intelligent tutoring systems according to student's knowledge. This approach, implemented in system called an Adaptive Courseware Tutor (AC-ware Tutor), has automatic generation of courseware elements, dynamic selection and sequencing of courseware elements, automatic generation of tests and questions, and it realizes adaptation to student's knowledge, as it's the most important feature. The content is adapted using the Bloom's knowledge taxonomy, Bayesian networks and students' stereotypes.

II. RELATED WORK

Among all of the systems that adapt to students' knowledge, we are interested only in those that automatically generate courseware: create, select, sequence and present courseware elements (for learning and testing knowledge). We could not find a system that has all desired elements, therefore, we have sought for the ones that are the most similar to our approach. We observed domain knowledge, student model, courseware generation, feature and level of adaptation and knowledge testing and closely analyzed only four systems that have the most features satisfied: DCG [6], ACE [7], ASM [8] and PAIGOS [9]. Even though, the mentioned references are not the newest one, recent approaches that relate to knowledge based adaptivity realization, in a form we were looking for, are not found.

Dynamics of selection and sequencing of courseware elements itself enables adaptivity of courseware, and most of these systems have dynamic selection and sequencing of courseware elements (DCG, ACE, ASM). The PAIGOS has static selection and sequencing with some indications of dynamics. None of the mentioned systems adaptively and automatically changes the actual learning content itself for each student individually. Since our focus is on automatization of adaptation, above mentioned solutions are not appropriate as they require enormous effort from teachers that “manually” create learning objects.The majority of systems claim to be adapted to student knowledge. However, this adaptation does not change the content of courseware elements, but refers to the selection and sequencing instead or to the presentation that requires tremendous “manual” work. It is crucial for adaptation to include the smallest possible courseware or domain knowledge granule that can be manipulated. In the mentioned systems, level of adaptation mostly relates to units, pages, lessons or tasks that include several domain knowledge concepts (ACE). Only in DCG and ASM level of adaptation relates to domain knowledge concept. Testing student knowledge is the "hot potato" in the e-learning systems. We were interested in whether the system itself generates questions (automatic) or teacher enters questions. Mentioned systems have no knowledge testing or the questions were created by teachers (DCG, ACE, PAIGOS).

III. AUTOMATIC, DYNAMIC AND ADAPTIVE COURSEWARE MODEL

In accordance with the reached findings and agreements, for the purpose of quality education realization, we have developed the idea of intelligent tutoring based on the model of automatic and dynamic generation and adaptive selection, sequencing and presentation of courseware. The new model of adaptive knowledge acquisition takes into account the current level of student’s knowledge that determines the complexity and level of presented courseware elements. The main feature of our approach is the automatization of courseware adaptivity in each stage of learning and teaching.

The adaptation described in this model relates to the domain knowledge concept, i.e. "atomic particle" of knowledge and in this sense indivisible. It enables automatic creation of learning objects based on the relations of domain knowledge concepts. The proposed model of adaptive knowledge acquisition is based on automatic and dynamic generation and adaptive selection, sequencing and presentation of courseware. Automatic courseware generation in the new model designates that the courseware elements for learning and testing knowledge are created by system itself. Dynamic courseware generation indicates that courseware is created in the moment of

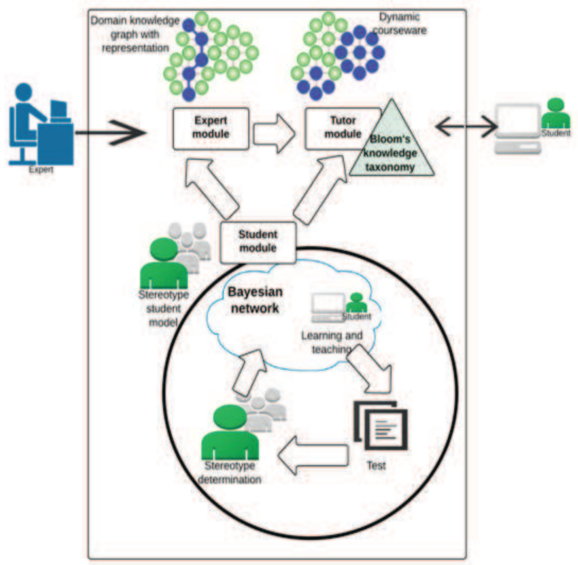

Figure 1. The structure of adaptive knowledge acquisition model

execution. Adaptive selection, sequencing and presentation of courseware are done automatically and dynamically in accordance with a student model using templates for statements and question.

The structure of the new model is shown in Figure 1. A student module is based on stereotypes defined according to the Bloom's knowledge taxonomy [10] and on Bayesian networks used to predict knowledge [11], as described in details in [12]. A tutor module is based on the Bloom's knowledge taxonomy and templates for statements and questions. The expert module [13] shows the relation between the concepts within the domain knowledge, which is the essential information that enables the tutor module to generate and present courseware. Courseware is made of courseware elements that contain a subset of domain knowledge. The tutor module sequences courseware elements to define the courseware structure and it is responsible for guiding the process of learning, teaching and testing knowledge. During knowledge testing, the tutor module cooperates with the expert module, generates questions and analyzes students' answers. Also, the tutor module uses the student model for planning further actions that affect the guidance of the process of learning, teaching and testing knowledge. The teacher module, that has the most changes compared to traditional intelligent tutoring systems architecture, is described in the nest subsection.

A. The tutor module

A courseware is an array of courseware elements that can be intended for learning and for testing knowledge. A learning courseware element is a subgraph of domain knowledge graph. Testing courseware element is a sequence of questions which test students' knowledge about the relations between concepts [12]. A learning courseware element is uniquely determined by its root and rank and it presents the knowledge object. The level of learning courseware element determines how large subset of the domain knowledge is included in that element.

1) Automatic generation of the learning courseware elements

Automatic courseware generation in the new model designates that all courseware elements are created by the system itself (not by the human teacher), based on the domain knowledge ontology. Generation of courseware elements starts from units, then for each unit its modules are generated, and lastly, for every module its lessons are generated. For each courseware element there is a specific generation algorithm.

2) Automatic and adaptive knowledge assessment

The prevalent problem that exists in computer assisted testing is necessity for the existence of an enormous number of questions of different difficulty level that are manually defined by teacher. It would be better to let the adaptive e-learning system itself to automatically and dynamically generate questions and adaptive tests based on the domain knowledge concepts. This kind of knowledge testing is used in the AC-ware Tutor model [12].

Testing courseware elements in the AC-ware Tutor contain questions that are generated over a domain knowledge subset. The input for knowledge testing is a subset of domain knowledge and the student stereotype. That subset of domain knowledge DK' corresponds to a union of learning courseware elements that come before the testing courseware element in the courseware CW.

Testing courseware elements contain certain number of questions generated using templates with four difficulty levels related to stereotypes and the Bloom's taxonomy. Each difficulty level examines a certain knowledge level. Questions are automatically and dynamically generated for each student separately, and therefore not repeated. Generated questions are rather rigid as they are created and corrected by computer tutor, not a human tutor. Both correct and incorrect answers in questions are selected based on the structure of domain knowledge graph.

We define two weight functions XA and XV on the domain knowledge graph whose values are determined after each knowledge test [12]. The value XV(Kx) represents the weighted sum of values of the function XA on the edges towards superconcepts of the concept Kx and towards subconcepts of concept Kx. The results of testing knowledge, that is, the values of weight functions, allow us to define stereotypes according to knowledge (see Table 1).

Before generating test, the current student stereotype has to be determined, what determines the level L of student's knowledge. Then for each two concepts that the student knows at a level equal to or less than L, one question, based on a template from difficulty category L+1, is generated (unless the student is an expert, then questions are generated based on the templates from the difficulty category 4). With this approach we can immediately determine whether there has been progress in the student knowledge (if a student can answer questions from the difficulty category that corresponds to the knowledge level higher that the knowledge level of student's current stereotype).

The concepts, that are included in the questions that the student has answered incorrectly, become an input for generation of questions based on templates from lower difficulty category L (unless L=0, because there is no

TABLE I. THE KNOWLEDGE STEREOTYPES

Stereotype	***The largest X_V values interval***	***Knowledge level***
Novice	$X_V(K_x)<0,2$	L=0
Beginner	$0,2\leq X_V(K_x)<0,4$	L=1
Intermediate	$0,4\leq X_V(K_x)<0,6$	L=2
Advanced	$0,6\leq XV(K_x)<0,8$	L=3
Expert	$0,8\leq X_V(K_x) \leq 1$	L=4

questions in difficulty category 0) and the concepts, that are included in the questions that the student has answered correctly, become an input for generation of questions based on templates from higher difficulty category L+2 (unless L=3 or L=4, because there are no difficulty categories 5 or 6). The test ends when there are no more questions that can be generated.

3) Adaptive selection, sequencing and presentation of the courseware elements

After defining the knowledge stereotypes and the way we consign them to students, it is essential to define how to select, sequence and present the courseware elements to students according to their stereotype.

Selection of the courseware elements is done according to the courseware elements level. The amount of knowledge that the student will learn in a single learning and teaching cycle is defined by the courseware element level. Therefore, lessons are selected for the stereotypes novice and beginner, modules for intermediate and advanced and units for experts. Testing courseware elements are added in the courseware according to rules that depend on particular stereotype [12].

Adaptive selection and sequencing of the courseware has to be followed by adaptive presenting of the courseware elements. The adaptive presentation is done in accordance with the Bloom's taxonomy. To be exact, the courseware element has to be presented to a certain stereotype in a form that corresponds its knowledge level. The stereotypes intermediate, advanced and expert knowledge are presented the same level knowledge, while the stereotypes novice and beginner are presented the higher level knowledge in order to enable much faster progress to intermediate stereotype.

Courseware element level determines the quantity of knowledge that the student will learn in a single learning and teaching iteration, while the Bloom's taxonomy defines a way of presenting knowledge to the student according to student’s stereotype. The templates for generating statements define the way in which the knowledge will be presented to student. Since we have defined four questions difficulty categories, we define the equivalent four statements difficulty categories [12]. Each statement category presents knowledge at a certain level.

B. The AC-ware Tutor effectiveness evaluation

The model AC-ware Tutor was implemented as a prototype version. The prototype uses domain knowledge "Computer as a System". To assess the effectiveness of the AC-ware Tutor, we have conducted an experiment carried out according to the pre-and-post test control group experimental design. Students who participated in experiment were undergraduate students from two faculties that took a course called „Introduction to Computer Science“. The experiment started in on 19th December 2012 and lasted until the 25th January 2013. At the very beginning of that experiment there were 205 students, but eventually only 158 of them completed all parts of the experiment (77%).

After a short introduction, during which the purpose of the experiment and general organizational issues were explained, the pre-test was conducted. Following the pre-test (OTI1), a brief introduction into organizational issues related to the treatments, was given. After the pre-test, we had to randomly divide students into the control and the experimental groups. Therefore, from 205 students that agreed to participate in the experiment, 98 were assigned to the control group and 107 were assigned to the experimental group.

The students from the experimental group used the AC-ware Tutor in learning and teaching process and the students from the control group was taught by live teacher in classroom. During the whole procedure, time slots, reserved for completing a certain step of the schedule, were identical for the experimental and the control group. After completing the learning and teaching process (2h per week, for two weeks) both groups performed the post-test (OTI2). Tests were used to measure the dependent variable – student knowledge. The test results were scored on a 0–100 points scale. As a final point, subjects from the experimental group got the chance to evaluate the AC-ware tutor by filling in a questionnaire, providing data on the subjective judgment of a teaching quality.

To be able to analyze results, it was important to find out the size of the student drop-off from each group. At the end of initial experiment, only 78 of 98 control group students and only 80 of 107 experimental group students completed all parts of the experiment.

Standard significance testing was used to investigate the effect of the treatments on the dependent variable. A null-hypothesis NH0 was stated: „There is no significant difference between the control and the experimental group“.

The descriptive statistics and results of KS test for the experiment are presented in Table 2. Columns OTI1 and OTI2 show calculated values for mean, median, standard deviation and variance of raw data collected during the pre-test and the post-test for both experimental and control groups. Column that starts with „Gain“ shows the calculated values for mean, median, and standard deviation of the differences between post-test and pre-test scores.

The results of statistical hypotheses testing are presented for hypothesis NH0. Table 3 shows the results of testing hypothesis NH0 using an F-test and two-tailed t-test for independent groups. After the initial experiment results' analysis, we have calculated the small effect size of 0.04. A confidence interval of the effect size is between -0.271 and 0.352. This interval includes zero, so we can say that the resulting effect size is not statistically.

Table 3 shows that null hypothesis NH0 has been accepted (no statistical significance). Since there is no statistically significant difference, the AC-ware Tutor can be used in a case, for example, the absence of the teacher, because the advantage of this system is the possibility of learning from home. Since this method of

TABLE III. DESCRIPTIVE STATISTICS FOR THE EXPERIMENT

	OTI1	*OTI2*	*OTI2 - OTI1*
Control group (78 students)			
Mean	66,15	71,05	4,91
Median	63,79	74,00	4,75
Standard deviation	17,05	14,31	12,88
Variance	290,61	204,67	165,94
KS D (Crit. KS α=0.05 is 0,15)	0,108	0,086	0,06
Experimental group (80 students)			
Mean	67,06	72,49	5,43
Median	63,39	73,50	4,89
Standard deviation	16,29	15,52	12,15
Variance	265,31	241,01	147,70
KS D (Crit. KS	0,13	0,108	0,078

TABLE II. RESULTS OF TESTING HYPOTHESIS NH0

	OTI1	*OTI2*	*OTI2 - OTI1*
F-test			
F-test (Crit. F α = 0.05 is 1,45) df (79 exp., 77 contr.)	1,09	1,18	1,12
p-value	0,34	0,24	0,30
t-test			
t-value (Crit. t α = 0.05 is 1,97)	0,36	0,60	0,26
df=(80-1)+(78-1)=156			
p-value	0,73	0,55	0,79
Effect size Δ	-	-	0,04
Confidence interval			-0.271 to 0.352

learning and teaching is not a replacement for the traditional learning and teaching process, we recommend a combination of both. The results of the research were affected by the "frivolity" of students, because they did not take these assignments seriously, and they have spent very little time on acquiring the domain knowledge. The students who have shown an interest in the learning, did not have a problem, and gradually progressed through stereotypes.

At the end of experiments, participants in the experimental groups had the chance to make comments or improvement suggestions, and could raise issues or problems that they encountered during the treatments, in the questionnaire. Apart from some improvement suggestions related to technical aspects of the system usage, comments mainly supported the findings of the quantitative analyses. Negative comments mainly addressed the difficulty of understanding the structure of the domain knowledge. Two thirds of the students said they want to continue to learn with the help of the AC-ware Tutor. Most students (70%) understood partly the contents of the AC-ware Tutor. The most students find that using of the AC-ware Tutor would be helpful supplement to traditional learning and teaching process. About 80% of the students considered that the quality, the functionality and the clarity of the AC-ware Tutor is average or better than average. On the basis of this questionnaire it can be concluded that the students generally understood computer designed course content processed with the help of the AC-ware Tutor and generally would like to continue to use the AC-ware Tutor in teaching.

IV. CONCLUSION

The idea of a new model of an intelligent tutoring system is based on traditional architecture, but with a substantial improvements that are associated with adapting the whole process of learning, teaching and testing to current level of student’s knowledge. In this regard, adaptation to student’s knowledge is achieved by applying stereotypes, Bayesian networks and the Bloom's knowledge taxonomy. The main feature of our approach is the automatization of courseware adaptivity in every stage of learning and teaching process.

The empirical study presented in this paper investigated the effect of using the AC-ware Tutor. Although the results are promising, we expected to get larger effect size. A reasonable explanation for the small could be that the presentation of domain knowledge in the AC-ware Tutor is rather novel for students and therefore difficult to grasp and apply in earlier phases of the experiment. When students get familiarized with the system's knowledge presentation, the system itself is expected to become very efficient. As a consequence, in future experiments, the presentation of the AC-ware Tutor should be improved.

It should be emphasized that the presented exploratory research is just the first step of a series of experiments, which - after the modification of the AC-ware Tutor based on the comments in the questionnaire - might yield more generalisable results in the future. Results gained through the conducted experiment have shown a need for adding some extended functions for courseware development and learning management in the AC-ware Tutor in order to get it as close as possible to the Bloom's 2-sigma target. In this sense, the system needs a graphical component that would facilitate the adaptation of courseware. Furthermore, since the ontology allows multilingual naming of concepts and relations, translation of templates for generating statements and questions would enable creating multi-language version of the AC-ware Tutor.

ACKNOWLEDGMENT

This paper describes the results of research being carried out within project 177-0361994-1996 *Design and evaluation of intelligent e-learning systems* within the program 036-1994 *Intelligent Support to Omnipresence of e-Learning Systems,* funded by the Ministry of Science, Education and Sports of the Republic of Croatia.

REFERENCES

[1] D. Sleeman and J.S. Brown "Introduction: Intelligent Tutoring Systems: An Overview", Intelligent Tutoring Systems (Sleeman, D.H., Brown, J.S.), Academic Press, Burlington, 1982, pp. 1-11.

[2] D. Benyon and D. Murray "Adaptive systems: From intelligent tutoring to autonomous agents", Knowledge-Based Systems, 6(4), 1993, pp. 197-219.

[3] J. Lee and O. Park "Adaptive instructional systems", In J. M. Spector, M. D. Merrill, J.V. Merriënboer and M. P. Driscoll (Eds.), Handbook of research on educational communications and technology (3rd ed.), 2008, pp. 469–484, New York: Taylor Francis

[4] C. Mulwa, S. Lawless, M. Sharp, I. Arnedillo-Sanchez and V. Wade "Adaptive educational hypermedia systems in technology enhanced learning: a literature review", ACM Special Interest Group for Information Technology Education Conference (SIGITE), 2010

[5] P. Mohan, J. Greer and G. McCalla "Instructional planning with learning objects", Knowledge Representation and Automated Reasoning for E-Learning Systems, 2003, pp. 52-58.

[6] J. Vassileva "Dynamic CAL-courseware generation within an ITS-shell architecture", Computer Assisted Learning, 1992, pp. 581-59.

[7] M. Specht and R. Oppermann "ACE-adaptive courseware environment", The New Review of Hypermedia and Multimedia, 4(1), 1998, pp. 141-161.

[8] P. Karampiperis and D. Sampson "Adaptive learning resources sequencing in educational hypermedia systems", Educational Technology and Society, 8(4), 2005, pp. 128-147.

[9] C. Ullrich "Pedagogically founded courseware generation for web-based learning: an HTN-planning-based approach implemented in PAIGOS", Lecture Notes In Computer Science, Vol.5260,232, 2008

[10] B.S. Bloom "Taxonomy of educational objectives. The classification of educational goals, Handbook I Cognitive Domain", New York, Committee of College and University Examiners, Longmans, 1956.

[11] Z.A. Pardos and N.T. Heffernan "Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing", International Conference UMAP 2010, pp. 255-266.

[12] A. Grubišić, S. Stankov and B. Žitko "Stereotype Student Model for an Adaptive e-Learning System", Special Journal Issue on Advances in Information and Intelligent Systems (Vol.76). International Conference on Information and Intelligent Systems, 2013.

[13] A. Grubišić, S. Stankov and I. Peraić "Ontology based approach to Bayesian student model design", Expert Systems with Applications, Vol 40(13), 2013, pp. 5363–5371