Evaluating the instructor support provided by a Web-based authoring tool for building adaptive courses

Maria Moundridou, Maria Virvou
Department of Informatics, University of Piraeus
mariam@unipi.gr, mvirvou@unipi.gr

Abstract

WEAR is an authoring tool for building Intelligent Tutoring Systems (ITSs) over the Web. The innovative feature of WEAR is its instructor modelling capabilities. In particular, WEAR maintains instructor models for the human instructors who author the ITSs in order to adapt the interaction with each instructor and provide individualised support to them throughout the ITS's lifecycle. The instructor modelling component may then be used to promote the collaboration among instructors who share the same interests. In this paper, we report on an evaluation study that we conducted in order to examine the necessity and added value of the existence of an instructor modelling component in WEAR's architecture. The results of the study are in favour of this component. Almost all instructors found useful to consult other peers through sharing the design of the course in order to redesign their course for future use. Furthermore, they showed interest in seeing the courses that were created by expert peers rather than courses created by novice ones.

1. Introduction

In the last decade a large number of researchers in the area of computer-based learning have put their efforts in the design and development of authoring tools for Intelligent Tutoring Systems (ITSs). The reason for this increased interest is that authoring tools may provide environments where instructors may easily author their own ITSs in varying domains. In this way, painful constructions of ITSs, which may not be reusable, are avoided. A special category of ITS authoring tools is the one dealing with the construction of adaptive Web-based textbooks. An adaptive Web-based textbook is an educational hypermedia application over the WWW, which customizes its content, style, and/or sequencing of the teaching material to the needs of the particular student. The main adaptive hypermedia technologies that are being used are twofold [2]: (i) adaptive presentation, where adaptation is performed at the content level and (ii) adaptive navigation support, which is performed at the link level. Examples of authoring tools that can be used to generate Web-based textbooks providing adaptive navigation support and/or adaptive presentation are Interbook [4], AHA [5], MetaLinks [8] and WEAR [7], which is the system under investigation in this paper.

However, even through using an authoring tool, the design of adaptive Web-based courses by instructors is not necessarily easy for prospective authors. Instructors may face several difficulties during the design process; for example, they may not be certain about the structure that their course should have. Another problem may occur when instructors provide inconsistent information to the tool that may lead to the generation of adaptive textbooks with problematic behaviour. For example, they may define prerequisite relationships in such a way that some sections of the textbook would be unreachable or never recommended. Furthermore, in order to be domain independent and generic, most authoring tools embody predefined pedagogical rules which cannot even be configured; instructors are obliged to accept them even if these rules contradict the way they perceive instruction.

To overcome some of these problems, several approaches were proposed. For example, Brusilovsky [3] introduces a concept-based course maintenance system which can check the consistency and quality of a course at any moment of its life and assist the course developer in some routine operations. A more sophisticated approach is presented by Nkambou et al. [9]: In order to provide designers with support that focuses on the expertise for building courses, they propose to use an expert-based assistant integrated with the authoring environment. The expert system reasons on a constraint base that contains constraints on curriculum and course design that come from different instructional design theories. In that way, the expert system validates curriculums and courses produced with the authoring tool and advises the instructional designer accordingly.

WEAR's approach for providing support to the instructional designers is mainly based on the instructor modelling component, a special component incorporated in WEAR's architecture [11]. The instructor models hold information about each instructor's preferences and interests. This information is used by the system to tailor the interaction with each instructor and promote the collaboration among those who share similar interests.

In an empirical evaluation study that we conducted, we explored whether the support that WEAR provides to

instructors can lead to the generation of better and more efficient ITSs and whether the instructor modelling component that WEAR embodies and relies on really helps in this direction. The study has mainly focused on the support of instructors in the generation and maintenance of adaptive textbooks. This facility of WEAR is completely domain-independent and thus is addressed to a very wide range of instructors of very different backgrounds.

This paper describes the evaluation study and discusses the outcomes. In general, the results of the study seem to be very positive and encouraging. In the subsequent sections we will briefly describe how WEAR is operating for the instructors (authors) of the course; we will then present the aims and settings of the conducted study and its results; finally, we will give the conclusions drawn from this work.

2. Authoring with WEAR

Authoring an ITS with WEAR involves creating and structuring the teaching material, constructing problems and tests and managing the student records. This section focuses on the creation and structuring of the teaching material which results in the construction of textbooks, since this is mostly related to the subject of the evaluation study presented in this paper.

Although most of the existing authoring tools for adaptive educational textbooks approach the adaptivity issue in quite similar ways, they differ a lot in the authoring process that their users have to conform with. For example, in Interbook [4] the author should provide a specially structured, annotated MS-Word file. In AHA [5] the author should write annotated HTML files. MetaLinks [8] on the other hand, provides a GUI interface for authoring all aspects of the electronic textbook. In WEAR, authoring is addressed in a way that in its first steps resembles a simple process adopted by commercial tools like WebCT [6]. It should be noted that this similarity between WEAR and WebCT concerns only the authoring procedure and not the resulting courses which in the case of WebCT are not adaptive.

In WEAR, instructors are assisted in authoring a course along various dimensions. First of all, when building the course they are provided with tools that verify the consistency of the course and report possible problems or errors. For example, an author is informed about cases where the prerequisite relationships impose that a topic indirectly requires the knowledge of itself. To offer more intelligent and individualised help to instructors WEAR relies on the information provided by the instructor modelling component that it embodies as well as on the information provided by the domain and learner models. In the subsequent paragraphs we will briefly describe the structure and use of these models.

However, for an extensive description of WEAR's user models the reader is referred to [10].

The *domain model* containing knowledge about the subject matter is structured as a network of hierarchically organised topics which constitute the textbook sections. Links between nodes of that network represent relationships between topics. At the moment, two types of relationship are used: *is_prerequisite_of*, to describe a topic a learner should know before accessing the more advanced one, and *is_related_to*, to describe that these two topics are in some way related to each other. Each topic has an associated *difficulty level* ranging from 1 (very easy) to 5 (very difficult). Finally, problems and/or tests examining the knowledge that must be acquired by studying a particular topic are associated with this piece of knowledge; these associations are also part of the domain knowledge.

For each topic contained in the domain model, the individual *learner model* stores two attribute-value pairs. These are: (i) *Read* (true or false) indicates if this topic has been visited by the student and (ii) *Knowledge weight* (ranging from 0 to 1) is an estimation of the student's knowledge level on this topic; it is calculated taking into account both the student's performance in solving the problems associated with this topic (if such problems or tests exist) and also the value of the *Read* attribute. The learner model is a combination of a stereotype and an overlay student model. The stereotype model (formed either directly by the instructor or after a preliminary test posed to the student) classifies initially the student according to his/her knowledge of the domain. As a result of this, each student is assigned to a stereotype (novice, beginner, intermediate or expert). The stereotype model also defines initial values for the overlay student model described above, taking into account each topic's *difficulty level*. If for example the stereotype model indicates that a student is "intermediate", then the initial value of the attribute *knowledge weight* will be 1 for all topics with difficulty level 3 or lower. The underlying assumption in this is that a student who is considered "intermediate" probably knows every topic which is not rated as difficult or very difficult. Each time a student visits a topic, solves a problem or completes a test, his/her user model is modified to reflect his/her current knowledge state in the domain being taught.

The *instructor model* is mainly used to support instructors in the authoring procedure. In particular, the authoring procedure is the following: The author should prepare HTML files for the topics that would be contained in the electronic textbook. The next step is to use WEAR's facilities for uploading these files to the WEAR server. For each uploaded file the author must also specify a title, a difficulty level and the position that it should have in the topics hierarchy. Finally, the author must define the *prerequisite* and *is_related_to*

relationships between topics. The author may also create multiple choice tests or problems and associate them with the appropriate topics. S/he may also create a preliminary test to be set to students in order to classify them in a stereotype and initialise their user model. In that case, the author should state which ranges in scores obtained from the preliminary test correspond to which stereotype. Alternatively, the author could manually define for each student of the virtual class the stereotype s/he belongs.

With regard to the adaptive textbook authoring, the *instructor model* mainly holds information obtained explicitly by the instructor. Such information is the instructor's long-term preference concerning the difficulty of the course. The instructor is also asked to specify how the students' level of knowledge will be calculated. For example, an instructor may state that s/he wishes the reading of a topic by a student to be given a weight of 20% and the rest 80% of the knowledge level to be obtained from the student's scores in problems and tests. Finally, instructors are requested to state their level of expertise in the domain being taught, their teaching expertise and their expertise in designing a course.

The information of the instructor model and the learner model is used by WEAR to support instructors in the authoring process in the following ways:

1. Instructors are offered the choice to see what other instructors have done. The information that is presented to the user in that case, is the structure of a similar course (in terms of the domain to which it belongs and in terms of the difficulty level assigned to it by its author) created by another instructor. In particular, the instructor may see an enriched Table of Contents presenting not only the topic hierarchy but also the *prerequisite* and *is_related* relationships between topics. In that way, instructors who may be novice as course designers could be assisted by more experienced peers who have used WEAR.

2. While students are working on the course, the system collects evidence to build reports and offer advice that may be of interest to the instructor. For example, if most students have a poor performance and the instructor's goal (as recorded in his/her user model) is to offer an easy course, then s/he is notified of the inconsistency.

3. WEAR also performs more thorough checks. For instance, if the majority of students fail to comprehend a specific topic (indicated by low scores in the corresponding tests), then the instructor is informed and is given some suggestions concerning this situation. For example, the instructor is informed that the underlying reason for the students' failure may be that s/he may have misplaced the specific topic in the curriculum, or otherwise it may be that the test was too difficult. By receiving feedback concerning the efficiency of the course they constructed, instructors can redesign it;

multiple iterations of this process may lead to the construction of optimal courseware for their class.

3. Aims and settings of the evaluation study

In an attempt to measure the value of the *instructor modelling* approach in supporting authors to generate and maintain their Web-based courses, an evaluation study was conducted involving nine instructors who were asked to build a course for MS-Excel using WEAR. MS-Excel was selected among other domains because it is usually taught by instructors of varying backgrounds and is addressed to learners whose backgrounds also vary significantly. All instructors participating in the experiment had a first and/or higher degree in Computer Science or Economics or Mathematics. In particular, the specific sample selected consisted of nine instructors with varying levels of expertise in teaching, in the domain of MS-Excel and in designing a distance learning course. All instructors were given 19 topics in MS-Excel and were asked to work with WEAR to build a beginners' course that should last two weeks. Before starting designing the course, instructors were introduced to WEAR's authoring and learning environment. Then, each instructor authored his/her course.

Two expert instructors with a teaching experience in MS-Excel of more than 500 hours were requested to comment on the nine instructors' course design. Henceforth, these two instructors will be referred to as commentators to distinguish them from the nine instructors who acted as authors in the experiment. In particular, for each constructed course the commentators pointed out the possible design "errors", such as the misplacement of one or more topics or the non optimal prerequisite relationships between topics.

Although this was only the first half of the experiment, the instructors' variety of design decisions provided substantial evidence that different instructors design the same course in different ways. If this was not the case and all courses were very similar, there would be no reason for having an authoring tool like WEAR to build ITSs; instead, one could develop the proper ITS only once without using an authoring tool.

Two weeks later, the nine authors were informed that they would receive feedback on their courses. WEAR was fed with the records of ten virtual students for each author. These simulated records were provided by the commentators. This is similar to a study conducted by Ainsworth et al. [1] who also used a virtual class instead of a real group of learners. Similarly to them, the reason why we opted for this approach was the fact that the comparison of multiple instructors' descriptions of a course would require them to author for the same group of learners. However, this would mean that the same

group of learners would have to operate multiple versions of the course, which was a non-acceptable option.

Authors were allowed to see their virtual students' history and progress in the course and they were also provided with a report pointing out the unsuccessful parts of the course. Instructors were then asked whether they would like to redesign the course if it was to be used again with a different group of students. For those willing to do so, WEAR provided two choices: either instructors could work by themselves or they could consult other peers. In the second case they could select the instructor, whose course they would like to see based on several criteria, such as the other instructors' level of expertise, or the performance of the other instructors' students, etc. This last session determined the degree of usefulness of the existence of the instructor modelling component in WEAR's architecture. If indeed instructors were interested in seeing what other more experienced peers had done, then it was useful to maintain instructor models that could provide information about instructors' expertise. Furthermore, if instructors were willing to redesign their course, then it would be useful to provide them with all relevant information in order to put their efforts in the parts of the course that needed to be redesigned.

4. Analysis of the results

To test whether different instructors design the same course in different ways the nine authors' structure of the course was compared. As it was expected and also acknowledged in similar studies (e.g. [1]), the structure of the nine courses differed a lot. The identified differences concerned both the prerequisite relationships between topics (e.g. Table 1) and the order in which the instructors wished the topics to be presented (e.g. Table 2). The fact that the nine instructors designed the same course in different ways shows that WEAR is indeed useful because it generates ITSs that may reflect the specific instructor's views for the course.

Table 1. Prerequisite topics for "Copying formulas" by instructors 1, 5 and 6

Prerequisite topics for "Copying formulas"	
Instructor 1	Getting started, Entering data in cells, Edit cell contents, Entering formulas, Selecting cells, Copying cells and data
Instructor 5	Getting started, Entering data in cells, Entering formulas
Instructor 6	Getting started, Entering data in cells, Formatting, Entering formulas

Table 2. Order of topics by instructors 1, 3 and 4

Instructor 1	Instructor 3	Instructor 4
Getting started	Getting started	Getting started
Entering data in cells	Entering data in cells	Entering data in cells
Edit cell contents	Automatically fill in data	Edit cell contents
Changing row height and column width	Changing row height and column width	Inserting cells
Entering formulas	Formatting	Deleting cells
Selecting cells	Edit cell contents	Copying cells and data
Deleting cells	Copying cells and data	Moving cells and data
Inserting cells	Moving cells and data	Changing row height and column width
Moving cells and data	Inserting cells	Entering formulas
Copying cells and data	Deleting cells	Copying formulas
Automatically fill in data	Selecting cells	Using functions
Copying formulas	Entering formulas	Paste function
Relative and absolute references	Copying formulas	Automatically fill in data
Formatting	Relative and absolute references	Formatting
Using functions	Using functions	Relative and absolute references
Paste function	Paste function	Selecting cells
IF function	IF function	Sorting data
Working with charts	Working with charts	IF function
Sorting data	Sorting data	Working with charts

The next question that sought an answer through the study was whether the support that WEAR provided to authors would be exploited by them in order to improve their course. The instructors' actions indicated that the answer to this question was definitely positive: eight out of nine instructors spent quite some time studying the "problems report" that WEAR provided and they did redesign their course.

The last issue that the study explored concerned the degree of usefulness of certain facilities that were based on the instructor modelling component that WEAR embodied. To reach a conclusion about this issue we explored the following aspects:

1. *Were authors interested in consulting other peers before redesigning their course? If so, in what way did they prefer to consult other peers?* The study revealed that all the instructors that redesigned their course (8 out of 9) took the opportunity to consult other instructors who had created a similar course. WEAR provided two choices to instructors wishing to consult their peers: they could either send them e-mail and ask whatever they would like to know, or they could consult them indirectly by seeing their course design. 6 out of 8 instructors preferred to see what the other instructors had done without sending any e-mail.

2. *Were there authors that were preferred as "consultants" over others?* Authors that were interested in seeing what other instructors had done could either select from a list of the available courses the one that they would like to see or they could select specific courses based on one or more criteria. All the authors preferred to search for the courses that satisfied some criteria. In particular, WEAR allowed instructors to select the course(s) that they would like to see based on: (i) the teaching expertise of the other instructor, (ii) the expertise of the other instructor in the domain being taught, (iii) the performance of the other instructor's students, (iv) the difficulty level of the course as defined by the other instructor. Each of the instructors made his/her own combination of criteria in order to see the other instructors' courses. However, a very common combination and in most cases (6 out of 8) the first that

was used was the one involving the criteria (i) and (ii). This showed that authors preferred to rely on peers that were both expert in the domain of the course and expert as teachers. Another extensively used criterion was the actual performance of the other instructors' class, which was related to the expertise of an author.

All the above mentioned findings lead to the conclusion that the instructor modelling component that WEAR embodies can play a crucial role in the generation and maintenance of Web-based courses. Almost all instructors found useful to consult other peers through sharing the design of the course. Furthermore, they did not like to do this blindly just by selecting to see a random instructor's course but they rather exploited the selection criteria that WEAR provided. The mostly preferred criteria were those concerning the other instructors' expertise. To offer this kind of information WEAR relies on the individual instructor models that it maintains.

4. Conclusions

WEAR is an authoring tool for building Intelligent Tutoring Systems over the Web. The innovative feature of WEAR is its instructor modelling capabilities. In particular, WEAR maintains a user model for the instructors who author the ITSs to be generated in order to adapt the interaction with each instructor and provide individualised support to them throughout the ITS's lifecycle.

This paper reported on a study conducted so that the utility of the instructor modelling component would be explored. Almost all instructors found useful to consult other peers in order to redesign their course for future use. Furthermore, they did not randomly select to see a course but they rather preferred to see the courses that were created by expert peers. Since the instructors' level of expertise is information that the individual instructor models hold, the conclusion we reach is that the instructor modelling component is a valuable part of WEAR.

The reported study in this paper was the first in a series of evaluation studies that we are planning to conduct in order to extend the scope and role of WEAR's instructor modelling component. Moreover, the fact that the results of this study were very encouraging towards the existence of this component in the architecture of WEAR shows the need for further research in the unexplored area of instructor modelling in ITS authoring tools.

5. References

[1] S. Ainsworth, S. Grimshaw, and J. Underwood, "Teachers implementing pedagogy through REDEEM", *Computers & Education*, 33, 1999, pp. 171-187.

[2] P. Brusilovsky, "Methods and techniques of adaptive hypermedia" User Modeling and User-Adapted Interaction, 6, 1996, pp. 87-129.

[3] P. Brusilovsky, "Course Sequencing for Static Courses? Applying ITS Techniques in Large-Scale Web-Based Education", in G. Gauthier, C. Frasson and K. VanLehn (eds.): *Intelligent Tutoring Systems, Proceedings of the 5th International Conference on Intelligent Tutoring Systems,* Lecture Notes in Computer Science, 1839, Springer, Berlin, 2000, pp. 625-634.

[4] P. Brusilovsky, J. Eklund, and E. Schwarz, "Web-based education for all: A tool for developing adaptive courseware", *Computer Networks and ISDN Systems,* 30, 1998, pp. 291-300.

[5] P. De Bra, and L. Calvi, "AHA: a Generic Adaptive Hypermedia System", in P. Brusilovsky and P. De Bra (eds.): *Proceedings of 2nd Adaptive Hypertext and Hypermedia Workshop*, Computing Science Report No. 98-12, Eindhoven University of Technology, 1998, pp. 5-11.

[6] M.W. Goldberg, S. Salari, and P. Swoboda, "World Wide Web - Course Tool: An environment for building www-based courses" *Computer Networks and ISDN Systems*, 28, 1996, pp. 1219-1231.

[7] M. Moundridou, and M. Virvou, "Authoring and Delivering Adaptive Web-Based Textbooks using WEAR", in T. Okamoto, R. Hartley, Kinshuk, and J.P. Klus, (eds.): *IEEE International Conference on Advanced Learning Technologies*, IEEE Computer Society, Los Alamitos, California, 2001, pp. 185-188.

[8] T. Murray, T. Shen, J. Piemonte, C. Condit, and J. Thibedeau, "Adaptivity in the MetaLinks Hyper-Book Authoring Framework", in C. Peylo (ed.): *Proceedings of the International Workshop on Adaptive and Intelligent Web-based Educational Systems*, Technical Report of the Institute for Semantic Information Processing, Osnabrück, 2000, pp. 61-72.

[9] R. Nkambou, C. Frasson, and G. Gauthier, "A new approach to ITS-curriculum and course authoring: the authoring environment", *Computers & Education*, 31, 1998, pp. 105-130.

[10] M. Virvou and M. Moundridou, "Student and Instructor Models: Two Kinds of User Model and their Interaction in an ITS Authoring Tool", in M. Bauer, P. Gmytrasiewicz, and J. Vassileva, (eds.): *User Modeling 2001: Proceedings of the 8th International Conference UM2001*, Lecture Notes in Artificial Intelligence, 2109, Springer, Berlin, 2001, pp. 158-167.

[11] M. Virvou and M. Moundridou, "Adding an instructor modelling component to the architecture of ITS authoring tools", *International Journal of Artificial Intelligence in Education*, 12, 2001, pp. 185-211.