Utilising the Student Model in Distance Learning

R. Alan Whitehurst
Department of Computer Science
3332 TMCB, PO Box 26565
Brigham Young University
Provo, UT 84602-6565 USA
1 801 378 5319
whitehur@cs.byu.edu

Christopher L. Powell
Department of Computer Science
3350 TMCB, PO Box 26565
Brigham Young University
Provo, UT 84602-6565 USA
bulls23@cs.byu.edu

Jason S. Izatt
Department of Computer Science
3350 TMCB, PO Box 26565
Brigham Young University
Provo, UT 84602-6565 USA
izatt@cs.byu.edu

1. ABSTRACT

A key to effective automated learning systems is the ability to dynamically adapt learning strategies to the needs of individual students. This presentation will describe the dynamic presentation facility of the DANDIE distance-based learning system that utilises a student model and intelligent system techniques to optimise presentation content and form to meet individual needs. Given a knowledge-base of course material, the system adapts the information through the selection of pre-defined learning strategies, dynamically constructs an interconnected set of presentation modules, and evaluates the effectiveness of the presentation strategy with respect to the individual student.

2. INTRODUCTION

We are living in an age of information. In the future, it is said that economic prosperity will be defined by a nations ability to produce knowledge workers. The development of an educated knowledge worker population is the goal and responsibility of educational systems, but in order to meet the demands for education in the coming century, several challenges relative to the quality and availability of education must be addressed. These challenges include: coping with an expanding population and increased demand for educational opportunities; providing quality educational experiences with a limited set of resources; decreasing individual cost to make educational advancement an option for more individuals; and expanding the opportunities for educational advancement to remote populations. Further, with the rate of scientific and technological change increasing exponentially, we must find an effective way to deal with knowledge obsolescence and the need for continually re-educating the workforce.

What is needed is a solution to the problem of servicing an expanding student body that is efficient and economical, maintains a high-quality of education, gives students flexible access to the collegiate experience, and has a manageable impact on the duties and activities of the faculty. We believe that recent technological advances in computer communications, networking, and in intelligent system theory provide a foundation upon which a "virtual" extension to the college campus can be developed based upon an integrated distance-learning paradigm that would meet the above criteria. Such a system could provide access to a quality of education currently unrealised in the university setting, while at the same time minimising the resources needed to meet the needs of an expanding student body.

In this paper we describe a system currently under development that supports the dynamic generation of web-based educational presentations in computer science. We will begin by giving a brief overview of the overall functionality and architecture of the system, and will then describe a set of prototypes that we have developed to illustrate these techniques.

3. DANDIE

DANDIE is an acronym that stands for Dynamic Asynchronous Networked Delivery of Individualised Education. Development of the DANDIE system was motivated by a desire to effectively educate large numbers of undergraduate students on the use of computers. It is designed around the concept of dynamic generation of course presentations in a manner that is optimised to meet the needs of the individual student. DANDIE extracts information from a domain model and applies specific pedagogical techniques in order to present the information to the student in the form of dynamically generated web pages. As the student interacts with the learning system

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

ITiCSE '98 Dublin, Ireland

© 1998 ACM 1-58113-000-7/98/0008... $5.00

through the browser interface, the system builds a model of the student's progress and attempts to correlate the effectiveness of the student's learning with the specific pedagogical techniques the system has used in generating the presentations thus far. The student model is then utilised to tailor the content and form of future presentations to the needs of the student. This capability is realised by utilising an internal storage format for course content that is based in first-order logic. This logical representation is presentation neutral, can be manipulated by intelligent programs, and can be transformed dynamically into presentations. Dynamic generation of course content allows the system to utilise the student model to influence what and how information is presented. Such individualised instruction surpasses the level of attention most educators are able to give students in crowded lower-level classes.

Facilitating knowledge acquisition and knowledge adaptability are key elements of the system we are developing. Through the use of intelligent authoring tools, developed specifically for the needs of faculty and targeted to the distance-based learning environment, course development is greatly simplified. Such tools would include: knowledge elicitation tools; course design agents for guiding the development process and detecting inconsistency or incompleteness; and smart editors for the creation of graphics, audio clips, and animation. Intelligent assistants, good interface design and complete documentation should make it easy for all instructors to capture knowledge within the framework of the distance-based learning system. Adopting this strategy has several additional benefits. First, developing a distance-learning system that utilises a logical formalism for knowledge storage will facilitate interfacing the system to digital libraries and other future knowledge repositories. Second, dynamically creating the presentations based upon intelligent transformation technology will allows the system to be targeted towards different languages and cultures by defining new sets of templates and transforms, thereby achieving a high-level of flexibility and adaptability at very low costs.

In the following sections, we will examine specific components of the DANDIE architecture, and explain how they work together to accomplish the goal of individualised instruction. These components include: knowledge elicitation tools and the domain knowledge module, the student model, the structuring and presentation agents, the student evaluation and performance agent, and the system performance agent.

4. KNOWLEDGE ELICITATION

Knowledge elicitation is necessary to populate the knowledge base from which DANDIE will draw course material. Traditionally knowledge acquisition has used a knowledge engineer as the intermediary between the domain experts and the computer system. We are exploring automated knowledge acquisition to reduce the role of the knowledge engineer and to more perfectly capture the expert's understanding. We understand that the expert would need initial training in the tool's use, so perhaps the role of the knowledge engineer would be reduced to simply providing this training. The expert's knowledge is stored in the domain knowledge model in a presentation neutral format. Utilising a logic with well-formed semantics allows the knowledge elicitation agent to be an active participant in the development of the domain model. Thus as new knowledge is entered, the knowledge model can continuously be automatically validated and checked for internal consistency.

5. THE STUDENT MODEL

One of the most formidable tasks facing educators is shaping their presentations of core knowledge to meet the individual needs of learners with varied and diverse cognitive and psychological traits. One way to provide individualised instruction is to maintain a model of each student. The student model is the means of determining the current knowledge state of the student (VanLehn, 1988). For the DANDIE system, the student model consists of three types of information: 1) what the student knows, 2) what the student has seen, and 3) the student's learning profile. The ratio of student knowledge over knowledge presented represents the effectiveness of the student's learning activities. Along with what information was presented, a record is also kept of which pedagogical strategy was employed for each of the information elements; this allows the system to infer the optimal teaching strategies for the student.

6. BUILDING THE STUDENT MODEL

In building the student model we've identified three necessary modules: interface module (interaction and presentation), evaluation module, and the storage module. The interaction/presentation facet of the DANDIE system is provided by the teaching agent and the student performance agent. The teaching agent is comprised of the structure agent and the presentation agent. Course material is determined by the structure agent, while format is determined by the presentation agent. In the process of structuring the information, the system utilises dynamic pedagogical strategies, such as the dynamic generation of analogies (Falkenhainer 1989, Whitehurst 1995), or the generation of concept summaries. The resulting knowledge content is passed to the presentation agent and formatted according to the student's preferences before delivery.

The student performance agent also encompasses the evaluation facet of modelling the student. Considerable research has been done in evaluating student progress. Although true natural language understanding remains elusive, some possible compromises exist (Riesbeck, 1991). Other methods include dynamic explanation generation (Acker, 1991), and differential modelling (Wilkins, 1988).

7. CURRENT PROTOTYPES

The prototypical systems developed to date have focused on individual parts of the system rather than a prototype of the entire product. A full application has been developed for knowledge elicitation as well as efforts in English text generation from the resulting logic. Two other applications explore, respectively, the choice of teaching paradigm based on Kolb's learning theory (Kolb, 1984) and using the student model for navigation through course material.

Knowledge elicitation is achieved using an automated tool. Graphical state machines are created by the user to represent procedural knowledge and declarative knowledge will be entered using a specification language. The expert also provides the names of state-defining predicates and specifies their values for each state. State transitions are defined by the user-provided actions and the changes that are caused in state-predicate values. Constraints in naming and predicate values result in a deterministic finite automata that can be automatically translated into first order predicate calculus. A predicate logic parser generates an abstract syntax tree from the textual logic which is subsequently stored in the knowledge base. The text generation module parses the AST, translating the logic into English text. This is the beginning of the dynamic presentation generation that will be such an integral part of the DANDIE system.

Conforming those presentations to proper student learning profiles is the key to customised education. This is done by utilising Kolb's theory of learning styles (Kolb, 1984) and the notion that a student can recognise or identify his/her own learning style. By asking appropriate questions about a student's learning preferences, the system can determine which of Kolb's learning styles, or combination therein, for which this student learns best. Since Kolb describes his styles as a circular continuum (Litzinger and Osif,1993), the potential for high granularity exists. That is, the model can support whatever degree to which the system can identify the students' learning style(s). Additionally, throughout the course, intermittent additions from other learning styles can be used to determine if the student has been diagnosed correctly or if refinements are needed.

Using the student model allows the student a certain freedom to explore the information, finding what interests him/her. Once the information has been extracted from the Domain Knowledge Model and conformed to the appropriate teaching strategy, the knowledge can be formatted into web pages and arranged in a temporary tree hierarchy knowledge structure. It will have basic principles build to more advanced topics following a depth-first approach going left to right. A dynamic navigation bar is generated with each page that allows the student to travel up the tree, to any child, or to any sibling page. The student is free to explore children and siblings, and whenever he desires to go back up the tree, the system insures that all information from that node and below has been viewed and mastered. If not, it sends back the appropriate page (unseen material, review, quiz, etc.).

8. REFERENCES

[1] Acker, Liane; Lester, James; Souther, Art and Porter, Bruce. 1991. Generating Coherent Explanations to Answer Students' Questions. In Intelligent Tutoring Systems: Evolutions in Design. 151-176.

[2] Falkenhainer, B., Forbus, K. D. and Gentner, D. 1989. The structure-mapping engine: Algorithm and Examples. In Artificial Intelligence, 41(1):1-64.

[3] Kolb, David A. 1984. Experiential Learning: Experience as The Source of Learning and Development. Englewood Cliffs, NJ: Prentice-Hall, Inc.

[4] Litzinger, Mary Ellen, and Osif, Bonnie. 1993. Accommodating diverse learning styles: Designing instruction for electronic information services. What is Good Instruction Now? Library Instruction for the 90s. 73-81.

[5] Riesbeck, Christopher K. and Schank, Roger C. 1991. From Training to Teaching: Techniques for Case-Based ITS. In Intelligent Tutoring Systems: Evolutions in Design. 177-193.

[6] VanLehn, Kurt. 1988. Student Modelling. Foundations of Intelligent Tutoring Systems. 55-78.

[7] Whitehurst, R. Alan, 1995. Systemic Software Reuse through Analogical Reasoning, Ph.D. Dissertation, Department of Computer Science, University of Illinois at Urbana-Champaign (TR95-1918)

[8] Wilkins, David C., Clancy, William J. and Buchanan, Bruce G. 1988. Using and Evaluating Differential Modelling in Intelligent Tutoring and Apprentice Learning Systems. In Intelligent Tutoring Systems: Lessons Learned. 257-277.