H. Ogata et al. (Eds.) (2010). Doctor Student Consortium Proceedings of the 18th International Conference on Computers in Education. Putrajaya, Malaysia: Asia-Pacific Society for Computers in Education.

Towards a Framework for Embedded ITSs

Sagaya AMALATHAS [1], Antonija MITROVIC[1], Ravan SARAVANAN[2] and David EVISON[1]
[1] *ICTG, University of Canterbury, New Zealand*
[2] *RANN Consulting, Malaysia*
sagaya.amalathas@pg.canterbury.ac.nz

Abstract: Although Intelligent Tutoring Systems (ITSs) have proven their effectiveness, very few attempts have been made to embed ITSs into existing applications. In this paper, we describe the research, design and progressive development of DM-Tutor (Decision-Making Tutor), the first constraint-based tutor (CBT) to be embedded within an existing system, the Management Information System (MIS) for palm oil plantation management currently being used in Malaysia. We discuss the research and development of DM-Tutor with the help of ASPIRE, an authoring system for CBTs. We also include future work planned for DM-Tutor.

Keywords: embedded intelligent tutoring systems, decision making training, management information system

Introduction

Intelligent Tutoring Systems (ITSs) provide benefits of one-on-one teaching to any number of students automatically and cost effectively through highly interactive environments [3]. Over the years many ITSs including LISP tutor [1], Andes-Physics Tutor [14], PUMP Algebra Tutor [4] and others have been effectively used in many teaching and learning domains. SQL-Tutor [6] and KERMIT [13] are among the many constraint-based tutors (CBT) [7] that have been developed and successfully implemented by the Intelligent Computer Tutoring Group (ICTG). Even though ITSs have been proven as effective teaching tools there have been very few attempts to embed them with other systems, Embedded Training System (ETS) [2], Excel Tutor [5] and Personal Access Tutor (PAT) [11]. Will ITSs embedded within existing systems provide effective instructions? Through this effort we hope to answer that important research question. With this research we aim to make several significant contributions. This will be the first attempt to embed a CBT with an existing system. We also aim to research the significant benefits of providing training through this integration. We aim to develop a framework for embedded ITSs and prove its research contribution through the development of DM-Tutor and its integration with the MIS for palm oil.

This paper presents the research, design and on-going development of DM-Tutor (Decision-Making Tutor), an ITS to train on plantation decision making for the palm oil domain. DM-Tutor will be embedded with a Management Information System (MIS) [10] currently being used to manage palm oil plantations in Malaysia and Indonesia. The MIS for palm oil contains operational data of yield records and plantation cultivation details. As the information contained is highly domain specific, managers who are new to the domain or to the MIS face difficulties in making accurate operational analyses and this affects the decisions they make at the palm oil plantations. When DM-Tutor is embedded with the MIS for palm oil, students will be able to practice plantation decision making using real life operational data from the MIS. The goal of DM-Tutor is to help students and managers apply theoretical concepts of plantation analyses into real-life plantation decision making. This paper is organized as follows. In the next section we describe DM-Tutor. Section two describes the development of DM-Tutor in ASPIRE [8], an authoring system for CBTs. Section three describes future work planned for DM-Tutor and conclusions.

1. DM-Tutor

To the best of our knowledge there has not been an ITS for plantation decision making, DM-Tutor is novel in that respect. Figure 1 presents the overall architecture of DM-Tutor. Our plan is to make DM-Tutor accessible through the MIS interface. The MIS by itself is a web based system and is accessed via a web browser. Students log in to the MIS and then access DM-Tutor through the MIS.

Student model contains information of student's knowledge of the domain and is updated every time the student uses DM-Tutor. Constraint based modeling (CBM) [9] is used to model the domain and the student. Student is modeled by looking through her/his solution and comparing it to the ideal solution in DM-Tutor. *Pedagogical module* selects instructions relevant to the scenario-based problem solving strategy used. It also has the role of providing helpful feedback to students when they submit an incorrect solution.

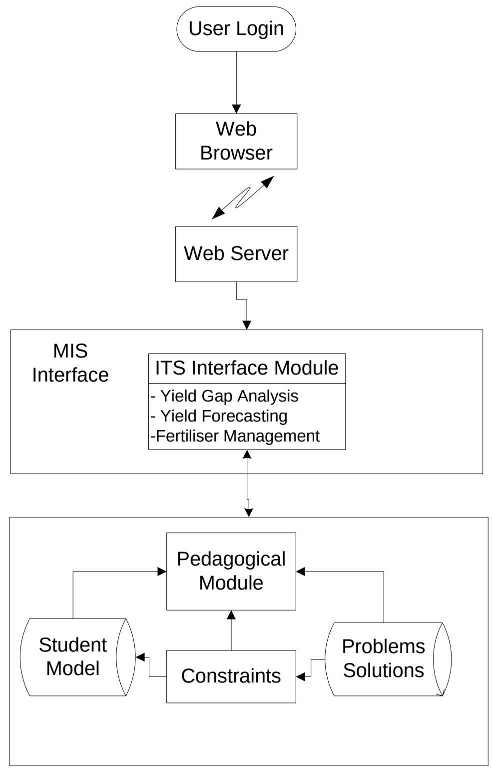

Figure 1: DM-Tutor Architecture

Interface module presents the student interface of DM-Tutor. The *problems* and *solutions* component focuses on *Yield Gap Analysis*, *Fertilizer Management* and *Yield Forecasting*, three main analyses of palm oil plantation management. In order to solve problems presented in DM-Tutor, students will have to access relevant reports from the MIS. DM-Tutor would also log information on the MIS reports that students accessed. This will enable DM-Tutor to provide helpful feedback when the student fails to provide correct solution to the problem.

2. Development of DM-Tutor

DM-Tutor has been developed using ASPIRE, an authoring system for constraint-based tutors. In the first development stage, we identified domain characteristics and problem solving steps required for DM-Tutor. To identify characteristics of the domain, we needed to determine the domain knowledge that we are teaching and the appropriate teaching strategy to be used. We divided the domain into three procedural tasks: *yield gap analysis*, *fertilizer management* and *yield forecasting*. In the next stage we developed ontologies for each separate task. Developing domain ontologies was a crucial step in DM-Tutor authoring process as this contributed to building of syntax constraints in the later development stage. In the domain ontologies we identified important concepts for each task and the relationship between these concepts using a hierarchical structure. Each concept could have super-concepts, sub-concepts, properties and relationships with other concepts. Concepts we created in the ontologies are present in the solution structures. By developing the ontologies we were able to study each task within the domain in greater detail and this helped in the development of accurate constraint bases.

In the third step we modeled the problem and solution structures for each task within the domain. For procedural tasks, problem solving is divided into several steps and student's solution is evaluated at every step. Students will not be able to continue to the next problem solving step before submitting the correct solution for the present step. Each solution component comes from the ontology of each task. In the next stage, ASPIRE developed the interface for DM-Tutor. Figure 2 presents the student interface for DM-Tutor. DM-Tutor's student interface is divided into three parts. The top pane is where the problem statement is placed, so that students always know the problem they are attempting. The bottom pane presents the solution workspace where students need to work on their solutions to the problems and the side pane is used to provide feedback to the students on the problems they are attempting. The interface design for DM-Tutor is aimed at reducing memory load of the students. We have planned to replace the current textual interface of DM-Tutor with java applets.

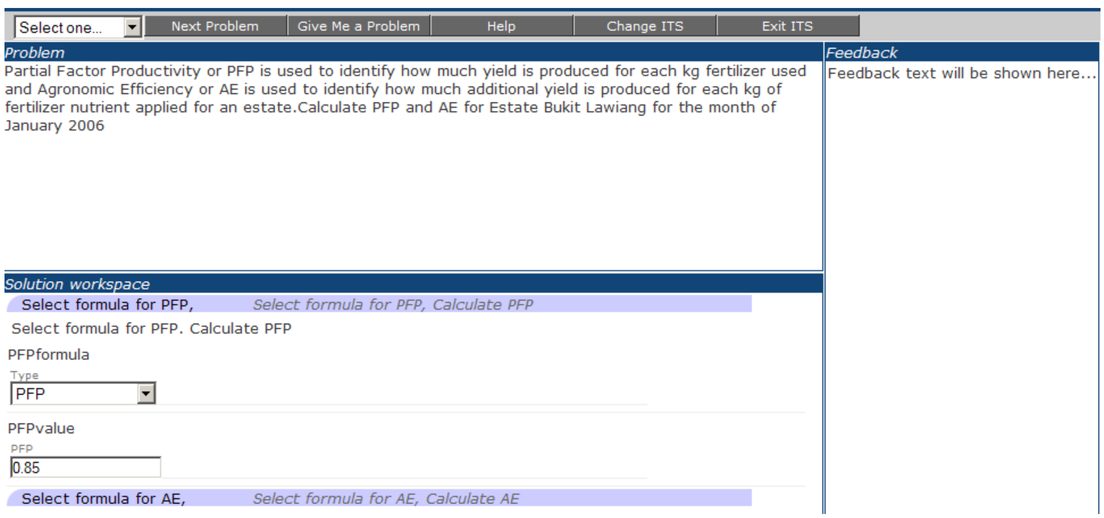

Figure 2: Student interface for DM-Tutor

In the next stage, ASPIRE helped to generate constraints for DM-Tutor. For CBM, knowledge of the domain is expressed as a set of constraints on correct solutions. ASPIRE generated syntax constraints and semantic constraints. Syntax constraints check whether the student's solution follows the syntactic rules of the domain. Syntax constraints were generated based on the concepts found in the ontologies. From the ontologies, there were 21 concepts in *yield gap analysis* task, 30 concepts in *fertilizer management* task and 26 concepts in *yield forecasting* task. Semantic constraints were generated based on the ideal solution for each problem. Semantic constraints checked whether student's solution matched the ideal solution. For DM-Tutor, ASPIRE generated 89 syntax constraints and 127 semantic constraints in total. After ensuring that all the information supplied for DM-Tutor was complete and consistent, we deployed DM-Tutor as an ITS.

3. Future Work and Conclusion.

At present we are researching possible ways to make an effective integration between DM-Tutor and the MIS. DM-Tutor will be accessed through the MIS. Students log in to MIS to attempt problems within DM-Tutor. For every problem in DM-Tutor, students would need to obtain relevant information from the MIS as part of their solution. We have planned to make information on the type of reports students looked at in the MIS to be send to DM-Tutor directly. This is to be done so that concise and specific feedback is provided to students when they submit an incorrect solution to a given problem. Currently I'm researching the efforts of Ritter and Koedinger [12], Cheikes, et al [2] and others to further understand and develop the most suitable architecture for embedding ITSs with existing systems.

Upon completion DM-Tutor will be evaluated by different user groups. We have planned for DM-Tutor to be evaluated by the Forestry Department, University of Canterbury. They will be looking at DM-Tutor as novice users. DM-Tutor will also be evaluated by plantation managers and trainee managers from TH Plantations, Malaysia and REAK Plantations, Indonesia. These users would be categorized as expert users for DM-Tutor. Through these evaluations we will be able to analyze how effective DM-Tutor is as a training tool. More importantly we would be able to analyze the benefits of embedding ITSs with existing systems in providing effective trainings.

Acknowledgements

The first author wishes to thank RANN Consulting for funding this research.

References

[1] Anderson, J. R. and Reiser, B. J. (1985). The LISP Tutor: It approaches the effectiveness of a human tutor. *BYTE 10, 4*. 159- 175.

[2] Cheikes, B.A., Geier, M., Hyland, R., Linton, F., Rodi, L. & Schaefer, H.P. (1998). Embedded training for complex information systems. *Artificial Intelligence in Education*, *10*, 314-334.

[3] Freedman, R. (2000). What is an intelligent tutoring system, *Intelligence, 11(3)*, 15-16.

[4] Koedinger, K. R., Anderson, J. R., Hadley, W. H., & Mark, M. A. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, *8*, 30-43.

[5] Mathan, S.A. & Koedinger, K.R. (2002). An empirical assessment of comprehension fostering features in an intelligent tutoring system. *6th International Conference: ITS 2002.LNCS 2363*, pp. 330-343

[6] Mitrovic, A. (2003) An intelligent SQL tutor on the Web. *International Journal Artificial Intelligence in Education, vol. 13, no. 2-4*, 173-197.

[7] Mitrovic, A., Martin, B., Suraweera (2007), P. Intelligent tutors for all: Constraint-based modeling methodology, systems and authoring. IEEE Intelligent Systems, special issue on *Intelligent Educational Systems, vol. 22, no. 4*, pp. 38-45

[8] Mitrovic, A., Martin, B. Suraweera, P., Zakharov, K., Milik, N., Holland, J., McGuigan, N. (2009) ASPIRE: An Authoring System and Deployment Environment for CBTs. *International Journal Artificial Intelligence in Education, vol 19, no 2*, 155-188

[9] Ohlsson, S. (1994), Constraint-based modeling. In Greer, J.E., McCalla, G (eds): Student modeling: the key to individualized knowledge-based instruction, 167-189

[10] Ravan, S. (2007). MIS for Palm Oil. RANN Consulting. http://www.rann-consulting.com

[11] Risco, S. & Reye, J. (2009). Personal Access Tutor-Helping students to learn MS-Access. *In Proceedings of the 14th International Conference on Artificial Intelligence in Education*. pp. 541-548

[12] Ritter, S. & Koedinger, K.R. (1996). An architecture for plug-in tutor agents. *Journal of Artificial Intelligence in Education, 7(3-4)*, 315-347

[13] Suraweera, P. & Mitrovic, A. (2004). An intelligent tutoring system for entity relationship modeling. *Artificial Intelligence in Education*, 14, 375-417.

[14] Vanlehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R. H., Taylor, L., Treacy, D. J., Weinstein, A., and Wintersgill, M. C. (2005). The Andes physics tutoring system: Lessons Learned. *Artificial Intelligence in Education*, 15, 147-204