Teaching Logical Database Design in ERM-Tutor

Nancy Milik, Melinda Marshall, Antonija Mitrovic

Intelligent Computer Tutoring Group
Department of Computer Science and Software Engineering
University of Canterbury, Christchurch, New Zealand
{nmi14,tanja}@cosc.canterbury.ac.nz

Abstract. We present ERM-Tutor, a constraint-based tutor that teaches logical database design (i.e. mapping conceptual to logical database schemas). Students practice this procedural task in ERM-Tutor by solving each step and receiving feedback on their solutions. We also present a new feature added to the system, which enables students to ask free-form questions. A preliminary evaluation carried out on ERM-Tutor investigated how students use free-form questions, and provided promising results. We plan to perform a bigger study in early 2006.

1 Introduction

Constraint-based tutors have been successful in a variety of domains, such as conceptual database design, database queries, data normalization, UML and language learning [2,5,6]. Building on successful work, we have developed ERM-Tutor, in which students practice the algorithm for mapping conceptual database schemas (i.e. ER diagrams) into relational schemas. The ER-to-relational algorithm [4] consists of seven steps, which map the ER components in the following order: 1) regular entities, 2) weak entities, 3) 1:1 binary relationships, 4) 1:N binary relationships, 5) M:N relationships, 6) multivalued attributes, and 7) *n*-ary relationships. Although the algorithm is well-defined and short, students typically find it hard to learn and apply consistently.

ERM-Tutor is a web-based system, the main components of which are the pedagogical module, problem solver, student modeler, session manager and user interface. The tutor also contains a set of problems and 121 constraints representing the domain knowledge. The problem-solving process is broken into seven tasks, corresponding to steps in the mapping algorithm, each task presented to the student on a separate page. The student has to complete the current task in order to move on to the next one. The student can request feedback at any time. The short-term student model consists of a list of satisfied and a list of violated constraints. This model is used by the pedagogical module to present feedback to the student. ERM-Tutor also maintains a long-term student model, which is used for problem selection.

2 Question Asking Module

Intelligent Tutoring Systems provide feedback on students' actions, but students do not always understand the feedback they receive. Therefore, it would be beneficial for students to be able to ask free-form questions at any time. ALPS [1,3] allows the student to ask any question, to which the system replies with a pre-recorded video clip. The results show that the rate of unprompted questions is lower than in the case of one-on-one human tutoring. Furthermore, half of the questions are not related to problem-solving, but are rather social interactions. Most of the remaining questions are performance-oriented, and not deep questions that would facilitate learning.

In this light, we added a question-asking module to ERM-Tutor. We defined 98 distinct questions, based on our experiences in teaching the mapping algorithm and our experience with other constraint-based tutors. These questions can be categorized into interface usage ("What does the button *Check Step* do?"), definitions of terms ("What is a foreign key?"), diagram notations ("How is an attribute represented in the ER-diagram?"), mapping regulations ("How is a relationship mapped?"), and deeper questions ("Why are the steps arranged in this order?"). The question database additionally includes a number of repeated questions that are phrased differently, resulting in a total of 182 questions. In contrast to ALPS, the answers to questions are textual.

The TFIDF (Term Frequency Inverse Document Frequency) vector weighting scheme [7] was chosen as the information retrieval mechanism, as is the case in ALPS. In our system, the questions are read from the database and separated into words. The weight of each question and word is calculated, and words are indexed in a hash table. When the student asks a question, the same calculations are applied to the query string: it is also broken-up into words and their weights are calculated. Each question is then allocated a query weight. Finally, the answer corresponding to the question with the highest query weight is returned to the student. To evaluate the subjective relevance of answers, students are encouraged to submit their ratings of answers; however, the system does not enforce it to avoid mode errors and distractions from the problem solving task.

3 Preliminary Evaluation

We performed a preliminary study of ERM-Tutor with students enrolled in an introductory database course at the University of Canterbury in 2005, in order to investigate the usage of free-form questions. 29 students logged into ERM-Tutor at least once, but five students used it for less than two minutes and so their logs were excluded from analyses. The average interaction time was under one hour (mean=54min, sd=63min), ranging from several minutes to 4.5 hours over several weeks. The number of sessions ranged from one to four (mean=1.67, sd=0.96). On average, students attempted 4.6 problems and completed 25% of them.

Only eight students asked questions, with a total of 24 questions submitted. The number of questions per student ranged from one to five. The questions can be categorized into task-focused (50%), definition-focused (8%) and phatic questions (42%). Task-focused questions ask directly for help solving the problem (e.g. "How could I solve this table?"). For instance, three students copied

the feedback messages, added a question mark at the end or a "How to" at the start, and submitted them as the questions. Definition-focused questions ask for definition of terms. There were only two such questions submitted: "What is foreign key?" and "What is multivalue?" Phatic questions establish a sense of social mood (e.g. "What is your name?", "How are you?" and "How do you answer questions?"). Excluding phatic questions, 14 questions were relevant for students' actions. Five of these questions were answered correctly, and for two of these, the students specified highest relevance. The answer could not be found for one question. The remaining questions received answers which were related to the query, but were not useful to students. This happened when the students did not formulate questions well, but instead copied a part of the feedback message, adding a question mark at the end (e.g. "Make sure the relationship is 1:1?"). We intend to enhance our question database with these questions.

4 Conclusions

The paper presented ERM-Tutor, a new constraint-based tutor that teaches the procedural task of mapping ER diagrams into relational schemas. We enhanced ERM-Tutor with a question-asking module, which allows the student to ask a free-form question, which the system processes and returns the answer with the highest relevance weight, using the TFIDF weighting scheme. Our preliminary study showed some evidence that students welcome the idea of asking free-form questions and confirmed the need for eliciting deeper questions. We are currently investigating various techniques to encourage students to use the module, such as prompting students to ask more questions and even suggesting a question to be asked based on their student model. We plan to conduct a full evaluation study of ERM-Tutor in March 2006.

References

1. Anthony, L., Corbett, A., Wagner, A., Stevens, S., Koedinger, K. Student Question-Asking Patters in an Intelligent Algebra Tutor. In: J. Lester, R. M. Vicar, Paraguacu, F. (eds) Proc. 7th Int. Conf. Intelligent Tutoring Systems, pp. 455-467, 2004.
2. Baghaei, N., Mitrovic, A., Irwin, W. A Constraint-Based Tutor for Learning Object-Oriented Analysis and Design using UML. In: C.K. Looi, D. Jonassen, M. Ikeda (eds), Int. Conf. Computers in Education ICCE 2005, pp. 11-18, 2005.
3. Corbett, A,, Wagner, A., Chao, C., Lesgold, S., Stevens, S., Ulrich, H. Student Questions in a Classroom Evaluation of the ALPS Learning Environment. C-K. Looi, G. McCalla, B. Bredeweg, Breuker, J. (eds) Proc. 12th Int. Conf. Artificial Intelligence in Education, pp. 780-782, 2005.
4. Elmasri, R, Navathe, S. Fundamentals of Database Systems. Addison Wesley, 2004.
5. Mitrovic, A., Mayo, M., Suraweera, P., Martin, B. Constraint-based tutors: a success story. In: L. Monostori, J. Vancza and M. Ali (eds), Proc. 14th Int. Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems IEA/AIE-2001, Springer-Verlag Berlin Heidelberg LNAI 2070, pp. 931-940, 2001.
6. Mitrovic, A., Suraweera, P., Martin, B., Weerasinghe, A. (2004) DB-suite: Experiences with Three Intelligent, Web-based Database Tutors. Journal of Interactive Learning Research (JILR), vol. 15, no. 4, pp. 409-432, 2004.
7. Salton, G., Buckley, C., Term Weighting Approaches in Automatic Text Retrieval. Tech. report #87-881 Computer Science Dept, Cornell University, 1987.