

Intelligent Functional Dependency Tutoring Tool

Jessica Enright, Ben Chu, Osmar R Zaïane

*Department of Computing Science, University of Alberta, Canada
zaiane@cs.ualberta.ca*

Abstract: Computer tools for education, particularly those that allow a learner to work through many examples of a problematic type of exercise, can be helpful to that learner's understanding. Functional dependencies and their use in finding candidate keys are an area with which learners often have difficulty in undergraduate database courses. In addition to helping the students, a learning tool could help the instructor if it collected information about learner attempts of the exercises. This work makes an effort to develop such a tool, helpful to both learners and educators.

Keywords: Intelligent tutoring system, data mining

Introduction

One of the fundamental concepts to teach in a database design course is the concept of relation decomposition, which consists of dividing relations (or data tables) into smaller tables in order to reduce redundancy, eliminate wasted storage and more importantly reduce anomalies or inconsistencies due to data updates. The central tool in producing these decompositions and refinement of the database in what is called normal forms, is the theory of functional dependencies, often called normalization theory [2].

Learners often find functional dependencies among the most challenging topics presented in undergraduate database courses. As with many other logical problems, repeated practical exercises seem to help learners master the material. This motivation is the major thrust in developing FDTutor, a web-based tutoring system for functional dependencies with self-evaluation and group assessment. Intelligent tutoring systems (ITS) can assist both learners and educators, and can fit into the overall learning experience in a number of ways [1, 3, 4]. FDTutor will assist both learner and educator and show that data mining can provide useful patterns to help assess the learning process.

The FDTutor System

An ITS for teaching functional dependencies should be helpful to both learners and educators. We have designed our system with this in mind.

The learner can use our system to both practice exercises, and to evaluate his or her relative strength in the class. In addition to providing exercises and hints, FDTutor also provides analytical tools to the learner. He or she can view information on his or her performance on exercises relative to the rest of the learners, in both number of exercises completed and quality of solution.

The educator can use the system's analytical tools to tailor his or her presentation of material. The system provides association pattern mining and sequential pattern mining. This gives to educator the opportunity to detect repeated patterns of actions on the part of the learners. If learners are repeatedly making the same types of mistakes, an educator can observe this and then modify lesson plans accordingly. The educator can also access a

particular learner's record of activity in the system, as well as various simple statistics for individual problems and sets of problems.

Roughly, the system can be divided into the front and back ends. The front end runs on the learner's local machine, and the back end on a server. The front end (client) comprises the learner's graphical user interface (GUI), a problem generator, and a problem solver. Even if the learner is temporarily offline, he or she can still solve new problems, as they are generated locally. The activities performed locally are uploaded to the server once the connection is established. The learner can attempt different problems and compare the different attempts in terms of time spent, number of steps used to solve the problem and the number of hints requested. Some visualization tools can also allow the learner to compare his or her performance with others who also attempted the same exercise. The exercises are classified into 3 classes: beginner, intermediary and advanced, and the learner automatically changes category based on performance in previous attempts. Each attempt is timed and points are cumulated. Progressively more insightful hints are provided by request but cost penalties for the total point accumulation of each exercise.

The back end comprises logging utilities, information storage and retrieval, and utilities for performing and viewing data mining analytics. The logging utility starts on the client and collects all timed activities performed by the learner, from the creation of exercises, the attempts performed, the hints requested, to the individual axioms used. Individual logs are uploaded to the server for storage in a carefully designed log which provides means for analysis of individual learner activities or the assessment of groups of learners using data mining algorithms and visualization tools.

Discussion

A full implementation and learner testing of an ITS for functional dependency exercises is valuable for both learners and educators. Data mining allows an educator to better understand any difficulty the learners are encountering, and enrich the learning process as required. The ability to complete an effectively limitless number of exercises, and use statistics for self-evaluation relative to the rest of the learner group would be invaluable to learners.

FDTutor will be used on a cohort of undergraduate students taking a database course in the Fall of 2008. The approach would be at a first stage to provide FDTutor to a group of students while another group would be taught traditionally, then at a later stage provide the tool to both groups, and compare the performance of both groups on either stages. The preliminary tests for the tool evaluation were positive indicating approval by potential users.

References

- [1] S. El-Khoury, E. Aïmeur, P.R. Richard and J.M. Fortuny. (2005). Development of an Intelligent Tutorial System to Enhance Students' Mathematical Competence in Problem Solving. In *Proceedings of E-Learn World Conference on E-Learning in Corporate, Government, Healthcare, & Higher Education*. (2042-2049).
- [2] M. Kifer, A. Bernstein and P. Lewis. (2006). *Database Systems, An application-oriented approach*, 2nd edition, NY: Addison Wesley.
- [3] A. Merceron and K. Yacef. (2003). A web-based tutoring tool with mining facilities to improve learning and teaching. In *Proceedings of 11th International Conference on Artificial Intelligence in Education*, (201-208).
- [4] A. Mitrovic. (2002). NORMIT: A Web-Enabled Tutor for Database Normalization. In *Proceedings of: International Conference on Computer Education*: (1276-1280).