Investigating students' self-assessment skills

Antonija Mitrovic

Intelligent Computer Tutoring Group
Computer Science Department, University of Canterbury
Private Bag 4800, Christchurch, New Zealand
Phone (64) 3 3642987 ext. 7771, fax (64) 3 3642569
tanja@cosc.canterbury.ac.nz

Abstract: Student modeling approaches predominantly focus on modeling student knowledge. For effective learning, however, it is necessary to teach students how to learn, as well as to provide support for learning domain knowledge. Recently, a number of projects focused on students' learning strategies, and initiated work on modeling students' metacognitive skills, such as self-explanation and reflection. This paper focuses on self-assessment as an important metacognitive skill. We present results of an initial study carried out in the context of SQL-Tutor, a system that helps students to learn a database language. We found that not all students are good at evaluating their own knowledge, and that their knowledge level is an important factor. The study is an initial step towards incorporating the meta-level into the existing student model in SQL-Tutor.

Introduction

In order to help students learn more effectively and efficiently, intelligent educational systems need to be able to monitor students' metacognitive skills, represent them in student models and provide support for further development. Studies show that improved metacognitive skills can be taught [3] and result in improved problem solving and better learning [2,5,8]. Self-explanation is a metacognitive skill that requires the student to explain a solution provided by the teacher/system, and has received the most attention so far [5]. A variant is explored in [2], when the student is asked to explain his/her own solution. In both cases, significant gains have been achieved by the students who explained the solutions. Knowing when a person needs help and what kind of help is appropriate has also been addressed in a recent project [1]. A study of how an inspectable student model encourages reflection on one's knowledge is presented in [4].

This paper focuses on self-assessment. If students are to learn, they also need to be able to critically assess their knowledge. This skill is important to identify topics that need attention, to assess the difficulty of the current problem, and to decide whether to abandon the problem or keep working on it. We report on a study of students' self-assessment skills in the context of SQL-Tutor, an intelligent educational system that teaches the SQL database language to university students. SQL-Tutor provides a facility for students to select problems on their own, which requires students to be able to evaluate their own knowledge. For details of SQL-Tutor please see [6,7]. In the next section we describe our experiment, and the final section presents the results of the data analyses and conclusions.

The Experiment

Assessing one's own knowledge is a difficult task. Our hypothesis is that students are not generally good at evaluating their knowledge. We propose that there are several factors that influence this ability, and expect student's knowledge to be one of the main factors for being able to critically assess one's knowledge.

In order to evaluate our hypothesis, we performed an experiment on SQL-Tutor, which was modified slightly to allow for data collection. We focused on situations when students abandon the current problem, and ask for a new problem. In such situations, we asked for a reason for abandoning the problem. Three possible replies were offered: the current problem is too easy, or too difficult, or the student wants to work on a problem of a different nature. The student is then asked to specify the type of the next problem. For this purpose, there were seven groups of problems, one for each clause of the SELECT statement, plus the *any clause* option.

The participating students were enrolled in an introductory database course at the University of Canterbury. The usage of the system was voluntary. The system was demonstrated in a lecture at the beginning of September 2000. Prior to the experiment, all students listened to two lectures on SQL. During the experiment, there were five additional lectures and five labs on the Oracle DBMS. The experiment required the student to sit a pre-test consisting of three multi-choice questions, worth seven marks. All students' actions were recorded in logs. The post-test consisted of three questions administered on paper seven weeks after the start of the experiment.

Results and conclusions

Out of 142 students enrolled in the course, 79 logged on to SQL-Tutor and sat the pre-test. We excluded the logs of nine students who attempted no problems. The students had two sessions on average, with a total time on task of 95.6 minutes. The minimal number of problems attempted per session was just one, while the maximum was 30, with an average of 6.65. The students managed to solve 1.5 problems per session, or a total of 10.26 problems. The mean on the pre-test was 4.02 (SD=1.52), while the post-test results were better, with a mean of 5.01 (SD= 1.24). The difference between the pre- and post-test results is statistically significant (t=-4.49, p=1.63E-05).

Out of 70 students, 25 had not abandoned any problems. The remaining 45 students abandoned 3.87 problems on average. Most often (59.39%) the students abandoned problems without attempting them. We divided the students into two subgroups. Students who scored above average on the pre-test were put into the *more able* group (63.16%), and the rest into the *less able* group.

Table 1 presents statistics for the two subgroups. The mean of the post-test is lower than the mean of the pre-test for the more able students, although not significantly. However, the less able group benefited more from the system than their more able peers. More able students tended to work longer with the system and solve more problems. The number of problems abandoned after zero attempts is almost identical.

Group	Pre-test	Post-test	Total time	0 attempts	Solved

More able	5.6 (0.75)	5.4 (0.94)	152.6	4 (2.3)	79%
Less able	2.91 (1.06)	4.86 (1.49)	115	3.95 (2.31)	68.75%

Table 1. Statistics for the two groups of students with different prior knowledge

Next, we analysed the reason for abandoning problems. Out of 165 abandoned problems, 57 (34.54%) were the problems from the more able, and 108 (65.45%) were from the less able group. Therefore, less able students were much more likely to abandon a problem. The distribution of answers to this question is given in Figure 1. Less able students thought that the problem was too easy more often than more able students, although the inspection of the sessions very often contradicts the reason they specified. More able students asked for a different type of problems more frequently.

An analysis of the second question shows that more able students are better at identifying the types of problems they need to work on (Figure 2). As we hypothesized, less able students are not good at identifying the kind of problem to work on next, and therefore they specify *any clause* most often (in 69.44% of the cases). More able students ask for hard problems (*group by* and *having*) much more often than the other group (35.08% and 14.04% compared to 15.74% and 6.48%).

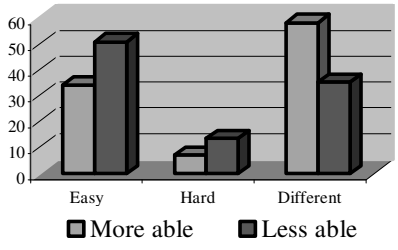

Fig. 1. Percentages of answers to question 1

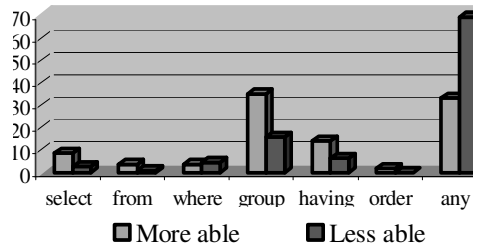

Fig. 2. Percentages of answers to question2

The results of data analyses justify our hypothesis. There are several ways to improve the SQL-Tutor system so that it may support students in acquiring self-assessment skills. The system may intervene in situations when a student keeps abandoning problems without trying to solve them, and encourage the student to solve the problem. Also, the system could intervene when the student does not have a preference about the type of the next problem. One way to help a student evaluate his/her own knowledge would be to visualize the student model. Since the student model in SQL-Tutor is quite complex, it could be summarized in a way similar to the answers offered for the second question. The student would then have a starting point from which to reason about their knowledge. Closer inspection of the student model may also have a positive effect on self-assessment skills. Future work on SQL-Tutor will develop the ideas presented in this paper further.

References

Aleven, V., Koedinger, K.: Limitations of Student Control: Do Students Know When They Need Help? In: Gauthier G., Frasson C., and VanLehn K. (eds): *Proc. 5th Int. Conf. ITS'2000*, Springer-Verlag, (2000) 292-303

Aleven, V., Koedinger, K., Cross, K.: Tutoring Answer Explanation Fosters Learning with Understanding. In: Lajoie, S.P., Vivet, M. (eds): *Proc. Int. Conf. AIED* (1999) 199-206

Bielaczyc, K., Pirolli, P., Brown, A.L.: Training in Self-Explanation and Self-Regulation Strategies: Investigating the Effects of Knowledge Acquisition Activities on Problem-solving. *Cognition and Instruction*, 13(2) (1993) 221-252

Bull, S. See Yourself Write: a Simple Student Model to Make Students Think. In: Jameson, A., Paris, C., Tasso, C. (eds): *Proc. 6th Int. Conf. UM'97*, Springer, (1997) 315-326.

Conati, C., VanLehn, K.: Further Results from the Evaluation of an Intelligent Computer Tutor to Coach Self-Explanation. In: Gauthier G., Frasson C., and VanLehn K. (eds): *Proc. 5th Int. Conf. ITS'2000*, Springer-Verlag, (2000) 304-313

Mitrovic, A., Hausler, K.: Porting SQL-Tutor to the Web. Proc. ITS'2000 workshop on *Adaptive and Intelligent Web-based Education Systems*, (2000) 37-44

Mitrovic, A., Ohlsson, S.: Evaluation of a Constraint-based Tutor for a Database Language. *Int. J. on Artificial Intelligence in Education*, 10(3-4), (1999) 238-256

Swanson, H.L.: Influence of Metacognitive Knowledge and Aptitude on Problem Solving. *J. Educational Psychology*, 82 (1990) 306-314