Data-Driven Worked Examples Improve Retention and Completion in a Logic Tutor

Behrooz Mostafavi(✉), Guojing Zhou, Collin Lynch, Min Chi, and Tiffany Barnes

Department of Computer Science, North Carolina State University, Raleigh, North Carolina 27695, US
{bzmostaf,gzhou3,cflynch,mchi,tmbarnes}@ncsu.edu

Abstract. Research shows that expert-crafted worked examples can have a positive effect on student performance. To investigate the potential for data-driven worked examples to achieve similar results, we generated worked examples for the Deep Thought logic tutor, and conducted an experiment to assess their impact on performance. Students who received data-driven worked examples were much more likely to complete the tutor, and completed the tutor in less time. This study demonstrates that worked examples, automatically generated from student data, can be used to improve student learning in tutoring systems.

Keywords: Worked examples · Data-driven · Problem-solving

1 Introduction and Related Work

In this paper we describe our study on the impact of data-driven worked examples in Deep Thought (DT). DT is a data-driven tutoring system for propositional logic that provides automatic verification of student proofs, and provides high-or-low proficiency problem sets at level intervals based on a student's performance [2].

Many intelligent tutors use pedagogical decision processes where the system is responsible for selecting the next action to take [4]. Pedagogical strategies are system-level policies that are used to decide what action to take when multiple options are available. We focus on one such decision: worked examples vs. problem-solving. When a student begins a new problem, the system can decide whether to ask them to complete it (*problem solving*); or provide them with a completed solution for review (*worked example*).

While numerous laboratory studies have shown the benefits of combining worked examples with problem solving [1], it is not always clear how they should be combined or how often each should be given. In a recent survey of the literature Najar et al. concluded that the research is still inconclusive on when worked examples should be given, how they should be scaffolded, and how they should be designed [3]. Consequently, most existing systems choose problem solving [5].

We investigated the impact of incorporating data-driven worked examples on student performance in DT. While worked examples involved in prior research

C. Conati et al. (Eds.): AIED 2015, LNAI 9112, pp. 726–729, 2015.
DOI: 10.1007/978-3-319-19773-9_102

were either hand created by domain experts or using expert solutions generated from expert systems, the worked examples in this study were extracted from prior student data. We hypothesize that the addition of data-driven worked examples will reduce the number of students who drop out of the tutor before completing all problems. We also hypothesize that the ordering of worked examples versus problem solving can have an impact on completion time and dropout. This is the first literature to investigate whether data-driven worked examples can be used to improve student performance.

2 Methods

DT was designed to add worked examples to the existing problem set. We derived a worked example for each problem in the DT student data corpus by selecting the shortest student solution that contained all of the logic rules that the particular problem was constructed to illustrate. Annotations for worked example steps were procedurally generated. In a worked example, students view sequential steps, one-by-one, until the complete solution has been constructed on the screen using data from our corpus. Students can move backward and forward between steps, using arrow keys, as needed.

Worked examples were assigned randomly on a per-problem basis. When students began a problem the system would decide whether to give a worked example or require them to work it themselves. This policy was balanced to ensure that the students worked at least one problem per level. An additional problem was added to the end of each level, which mirrored the rules and problem solving strategies as the other problems in the respective levels. This problem provides a built-in post-test on each level of the tutor, for comparison of performance of the same problem set with and without worked examples.

DT with data-driven worked examples was tested to determine its effect on tutor completion, student dropout, and how the order of worked examples affects student performance. DT was used as a mandatory homework assignment by students in two computer science discrete mathematics courses (WE group, $n = 261$). This data was compared to data collected from the previous version of DT with no worked examples for tutor completion and dropout comparison (NoWE group, $n = 47$).

3 Results and Discussion

Our first hypothesis is that data-driven worked examples would increase students' completion rate. Table 1 shows the number of problems solved, worked examples received, and total time spent in tutor for the WE and NoWE groups. Students in both groups solved the same number of problems in the tutor on average with no significant difference between them ($p = 0.327, power = 1$). However, the WE group spent 27% less time in tutor on average than the NoWE, even with the added worked examples. This was marginally significant ($p = 0.063$).

Table 1. The number of problems solved, number of worked examples received, and total tutor time for the WE and NoWE groups

Group	# Solved Problems			# Worked Examples			Total Tutor Time (mins)		
	Mean	*Median*	*StDev*	*Mean*	*Median*	*StDev*	*Mean*	*Median*	*StDev*
WE	14.12	13	5.09	7.53	8	1.35	224.4	97.7	346.5
NoWE	13.08	13	4.94	–	–	–	307.2	244.4	263.4

We used percent completion as a measure of overall success for each group of students. Percent completion is a measure of *how far* students progress through the tutor, on average. Table 2(a) shows the average percentage of tutor completion by group. Student dropout is defined as the termination of tutor activity at any point in the tutor prior to completing all of the assigned problems. It is a measure of *how many* students are using the system at any point in time. Table 2(b) summarizes the number of students who completed and dropped out of the tutor across both groups. Table 2(c) compares the retention trends for the WE and NoWE groups at the end of each level. From these figures, it is evident that the worked examples are dramatically improving retention and tutor completion.

Our results show that students using Deep Thought with worked examples completed more of the assigned problems and were less likely to drop out. We found a statistically significant difference between the groups in terms of percent completion (one-way ANOVA: $F(2, 308) = 6.38$, $p = 0.014$). The average percentage of tutor completion was significantly greater for the WE group than for NoWE (94% vs. 79.8%), an improvement of 14.2%. Thus adding worked examples to Deep Thought enabled students to complete more of the tutor. Additionally, the WE group was significantly more likely than the NoWE group to finish the tutor, as shown in Table 2(c). A much smaller percentage of the WE group dropped out of the tutor (9.6%) compared to the NoWE group (44.7%). Moreover the WE students who dropped out did so at a later point than the dropouts in the NoWE group. This indicates that the addition of worked examples increases the chances that students will be able to finish the tutor, and increases the percentage of problems they will complete.

Table 2. (a) Percentage of tutor completion by group. A * indicates significance. (b) Student completion of the tutor by group. Dropped indicates that the student did not complete Deep Thought. (c) Percentage of students remaining in tutor at the end of each level by group.

(a)	Mean	StDev
WE	94.02*	21.07
NoWE	79.79	29.88

(b)	Completed	Dropped	Total
WE	236 *(90.4%)*	25 *(9.6%)*	261
NoWE	26 *(55.3%)*	21 *(44.7%)*	47
Total	262	46	308

(c)	Level 1	Level 2	Level 3	Level 4	Level 5	Level 6
WE	96.2	95.0	93.9	92.7	92.3	90.4
NoWE	93.6	87.2	87.2	78.7	72.3	55.3

Our second hypothesis is that the ordering of worked examples and problem solving will affect student performance. We used the additional end-of-level post-test problems in DT3 to study the effect of the ordering of worked examples and problem solving practice in each level. We classified problem instances into two groups, *WE–PS* (where students viewed worked examples and then solved problems before the level's post-test problem) and *PS–WE* (where students solved problems, then viewed a worked example before the level's post-test problem). Our initial results show no significant differences between the *WE–PS* and *PS–WE* groups for step count and elapsed time. However, in levels 3 and 4 of DT, students in the low-proficiency track took significantly longer on each step in the post-test problem in both levels 3 and 4 if they saw worked examples before attempting to solve problems in these levels. These results indicate that there may be a disadvantage for low proficiency students to see efficiently-worked examples chosen from our corpus, however, further analysis is required. Therefore, our hypothesis that the ordering of worked examples versus problem solving would impact student performance was not confirmed.

4 Conclusions and Future Work

This study investigated the impact of data-driven worked examples on student performance in the Deep Thought logic proof tutor. Our results show that students who received with worked examples completed more of the required problems and were less likely to drop out of the tutor than those who only engaged in problem solving . Our results indicate that worked examples are very beneficial in Deep Thought. Our overall analyses do not show significant effects due to ordering of worked examples and problem solving across the tutor levels, however we found a difference in time per solution step in the ordering of practice problem types in levels 3 and 4 of the tutor, warranting further investigation. In future work, we plan to further investigate the impact of worked examples, and to apply machine learning to derive individualized pedagogical policies to select worked examples when it would be most beneficial for particular students.

References

1. Atkinson, R.K., Derry, S.J., Renkl, A., Wortham, D.: Learning from examples: Instructional principles from the worked examples research. Review of educational research **70**(2), 181–214 (2000)
2. Mostafavi, B., Eagle, M., Barnes, T.: Towards data-driven mastery learning. In: Proc. Learning, Analytics, and Knowledge (LAK 2015) (to appear, 2015)
3. Najar, A., Mitrovic, A.: Should we use examples in intelligent tutors? In: Proc. Computers in Education, pp. 5–7 (2012)
4. VanLehn, K.: The Behavior of Tutoring Systems. International Journal of Artificial Intelligence in Education **16**(2), 227–265 (2006)
5. Vanlehn, K., et al.: The Andes physics tutoring system: Lessons learned. International Journal of Artificial Intelligence in Education **15**(3), 147–204 (2005)