Combining Worked Examples and Problem Solving in a Data-Driven Logic Tutor

Zhongxiu Liu(✉), Behrooz Mostafavi(✉), and Tiffany Barnes

Department of Computer Science, North Carolina State University,
Raleigh, NC 27695, USA
{zliu24,bzmostaf,tmbarnes}@ncsu.edu

Abstract. Previous research has shown that worked examples can increase learning efficiency during computer-aided instruction, especially when alternatively offered with problem solving opportunities. In this study, we investigate whether these results are consistent in a complex, open-ended problem solving domain, where students are presented with randomly ordered sets of worked examples and required problem solving. Our results show that worked examples benefits students early in tutoring sessions, but are comparable to hint-based systems for scaffolding domain concepts. Later in tutoring sessions, worked examples are less beneficial, and can decrease performance for lower-proficiency students.

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

1 Introduction and Related Work

In this study, we focus on the pedagogical strategy of either presenting a logic proof problem to a student for completion (problem solving – PS), or providing a completed solution of the same problem for review (worked example – WE). Pedagogical strategies [9] are system-level policies that decide which action to take when multiple actions are available.

Worked examples are pedagogically beneficial, especially for inexperienced learners [3]. Interleaving worked examples and problems solving has been found to help students solve problems faster and more accurately on transfer post tests compared to blocked problem solving before worked examples [8]. Prior research has shown that Worked examples are more efficient and require less time on task than tutored problem solving. For example, McLaren and colleagues [4] found that replacing some tutored problem solving with isomorphic worked examples does not increase the learning effect but had significantly higher learning efficiency than problem-solving alone. However, in a more 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 [7]. Perhaps this is why most existing systems choose problem solving. In our prior work with the Deep Thought logic tutor, we showed that the addition of our data-driven worked examples reduced the time spent in

A. Micarelli et al. (Eds.): ITS 2016, LNCS 9684, pp. 347–353, 2016.
DOI: 10.1007/978-3-319-39583-8_40

tutor by 27 %, increased the amount of tutor completed by 14 %, and increased retention in the system by 35 % over students who were given problem solving opportunities alone [6].

In this paper we evaluate the impact of worked examples and problem solving opportunities on student performance in the Deep Thought tutor. We hypothesize that worked examples will reduce problem-solving time and decrease hint usage, but will not have an impact on the length of problem solutions or the percent of rule applications that are correct in first level problem solving. For later levels, we investigate the impact of the number of worked examples and their ordering with problem solving on overall performance in the tutor, as measured by time, number of hints, length of problem solutions, and rule application accuracy. This work will serve as the basis for future research on how and when we should use worked examples in data-driven problem solving environments.

1.1 The Deep Thought Tutor

Problem Levels and Proficiency Track: Deep Thought (DT) is a data-driven ITS for graphically constructing propositional logic proofs. DT presents proof problems consisting of logical premises and a conclusion to be derived using logic axioms. DT is divided into 6 strictly ordered levels of logic proof problems, each split into a high track with a few complex problems, and a low track with more simple problems. Level 1 is a single track, where students in the control group solve (S) three problems and the WE group solves 2–3 problems and receives 1–2 worked examples (E). At the end of each level, DT uses our data-driven knowledge tracing (DKT) system to assign students to the high or low track in the next level. This feature has been shown to reduce tutor dropout over versions of DT without problem selection or hints [5]. To ensure a fair comparison in this paper, we controlled for track placement in our analyses; this was not necessary in Level 1 where all students solve isomorphic problems.

Data-driven PS Hint and WE: DT utilizes data-driven hint generation via the Hint Factory, using prior student solutions to a problem to match problem states with new users, and giving hints that will guide students from their current state to the solution state [1]. The Hint Factory for DT leverages Interaction Networks constructed using prior student work to build a problem-specific domain model [2]. To create a worked example for a particular problem, we selected the shortest student solution in the Interaction Network that contained all the targeted logic rules for that problem. We then plug in information about the steps in the student solution into an annotation template, and present the WE step-by-step. Before deployment, experts checked our data-driven worked examples, to ensure their quality and correctness.

2 Methods

DT was used as in an undergraduate computer science class in Fall 2015. Course credit was awarded according to the number of levels completed. Students were randomly split into two groups; the control group (n=24) solved all problems, while the worked example (WE) group (n=51) viewed 1–2 of those problems as worked examples. For the WE group, the number and order of worked examples was chosen randomly in each level so that students received 1–2 worked examples (E) and solved (S) the final problem. Low track sequences are: EESS, ESES, SEES, ESSS, SESS, SSES; and high track sequences are: EES, ESS, SES.

To study our hypothesis that worked examples improve learning efficiency, we compared the performance of the control and WE groups on Level 1. In Level 1, the control group solved three problems and the WE group was randomly assigned to view 1–2 examples and solve 2–3 problems. To study the impact of the number of worked examples, we compared performance on problems solved by track and number of worked examples across Levels 2–6. We investigated the impact of order of practice by studying student performance on the last problem in each level for high track levels. Low-track levels were not large enough to compare the impact of order.

Measures include: time, rule-application accuracy, solution length difference, and the number of hints requested. Rule application accuracy is the percentage of correct rule applications out of all applications a student attempted. Solution length difference is the number of steps a student used over the shortest recorded proof for the given problem. This is a good measure of student problem-solving ability comparable across levels, as shorter solutions usually indicate more expert-like knowledge. Comparisons between groups were made using the Kruskal-Wallis test for one-way analysis of variance, since normality assumptions were not met.

3 Results and Discussion

In this section, we first present descriptive statistics for the WE group to demonstrate that they read the worked examples and solved the planned number of problems in each level. We then present three studies: comparison of the control and WE groups in level one, the impact of the number of worked examples on problem-solving in levels 2–6, and the impact of ordering for high-track WE levels. Times that were 3 or more standard deviations from the mean were considered outliers and were not included in the analysis. This resulted in a cap of 25 min for each solved problem, and 6 min for each example, excluding 174 of 1936 problem instances (8.98 %). We note that these cutoffs are necessary because students use DT through a web browser, and long times may indicate intense work or idle time that are not separatable in our data.

3.1 Time and Practice Type for WE Group

Students with worked examples received an average of 7.5 (SD = 1.09) worked examples, which accounts for 38.2 % (SD = 5.25 %) of the problems they encountered. WE students spent a mean of 10.02 % (SD = 15.17 %) of tutoring time on worked examples, and 5–10 sec on each step. We conclude that students are actually reading the worked examples, especially in earlier levels.

3.2 Comparison of Groups in Level 1

We then compared performance of the groups over all problems solved on Level 1. Hints were available for all 3 problems for the control group, but were not available for the 4th problem solved by the WE group. We found no significant difference between groups in learning time, hint usage, rule application accuracy, or solution length. Data for the WE group indicate that learning may have been more efficient for some students, but the variance in time, hint usage, and solution length difference was too high for this effect to be significant.

Table 1. Measurements for Level 1 in DT, by WE and control groups.

Level 1	Worked example group			Control group		
	Mean	Median	Std dev	Mean	Median	Std dev
Hints/problem	15.7	2.5	33.04	13.2	5.33	20.82
Total time (min)	33.18	26.64	23.32	35.28	29.1	22.83
Time/problem (min)	5.88	5.45	3.79	7.78	8.27	3.85
Avg PS step time (sec)	8.78	7.11	6.14	7.60	6.87	3.11
% correct applications	48.09	47.22	19.67	49.27	49.46	19.03
Solution length difference	2.93	1.33	2.32	3.17	2.0	2.15

From our data, the median time spent on worked examples by the WE group is under 1 min, for an average of 1.49 worked examples and 2.72 problem solved. This means that WE group solved almost three problems and viewed a worked example in about the same time that the control group solved three problems. This result confirms prior research that worked examples do not increase learning, but does not replicate the learning efficiency result. This may be also due to the availability of hints in DT. With the mean number of hints at 13 and 15, students in both groups clearly used bottom-out hints to generate step-by-step examples.

3.3 Number of Worked Examples

We further investigated the number of examples. In levels 2–6, the WE and control groups were not directly comparable, since our DKT assigned most WE students to the high track and control to the low track. This was because it assigned equal credit for actions in worked examples (viewing) and in problem solving (applying). Therefore, we aggregated data across levels 2–6 into groups by number of worked examples (0, 1, 2) and track. Table 2 reports central measures across all solved problems.

High-track levels with two worked examples have significantly shorter solutions than those with 0 or 1 examples. On the other hand, high-track levels with no worked examples had a higher proportion of correct rule applications. This,

Table 2. Measurements for high and low tracks in DT, by the number of worked examples encountered per level. $*$ indicates $p < 0.05$, † indicates $0.05 \leq p < 0.1$.

	High level track			Low level track		
# Worked examples	**0**	**1**	**2**	**0**	**1**	**2**
n	*49*	*186*	*57*	*80*	*35*	*16*
Hints/problem						
Mean	3.7	2.22	*NA*	2.22$*$	4.96$*$	4.18$*$
Median	0	0	*NA*	0.5	2.5	1.75
Std dev	10.7	8.36	*NA*	3.92	6.97	7.65
Solution length difference						
Mean	4.79$*$	5.08$*$	4.15$*$	4.3†	5.49	5.4†
Median	4.5	5	2	3.67	4.33	5
Std dev	3.17	3.62	3.84	3.02	3.32	2.78
% Correct applications						
Mean	0.76$*$	0.7$*$	0.7	0.7$*$	0.54$*$	0.54$*$
Median	0.81	0.7	0.71	0.7	0.55	0.54
Std dev	0.18	0.17	0.25	0.18	0.14	0.13
Average problem solving time						
Mean	386.78	376.30	338.21	244.71	242.89	299.51
Median	380.4	294.25	248.96	173.88	194.76	204.54
Std dev	266.60	252.77	290.55	181.63	213.01	217.45

along with short solutions, suggests that students with 0 or 2 examples may have more quickly learned a small set of rules to apply efficiently, while students with a single worked example may try applying more rules.

In the high-track, worked examples reduced dependence on hints. Surprisingly, low-track levels with no worked examples have significantly fewer hints and higher correct rule applications than low-track levels with worked examples. In this study, low track are those who have not demonstrated proficiency in problem solving, even with skill overestimation in the WE group. Therefore, we conclude that for low-proficiency students, worked examples increase both hint usage and the length of solutions. It may be that for DT, worked examples decrease self-regulation for low proficiency students solving simpler problems.

3.4 Worked Example Ordering

We hypothesized that interleaved PS practice and WE, with the PS occurring first, would result in better final problem performance as measured by time or length. In high-track Levels 2–6, we aggregated data based on the ordering of worked examples (E) and problem solving (S), where the possible orderings are EES, ESS, and SES. We studied only the high-track levels given the higher ordering and fewer students in low track. We found no significant difference for any performance measurements. This result corresponds with previous research that worked examples work as well as problem solving for learning.

4 Conclusions

To summarize, we found that the impact of worked examples may be complex and individual in environments for open-ended complex problem solving. The results of our Level 1 controlled study show no significant differences in problem solving time, solution lengths, accuracy of rule applications, or hint usage per problem. However, the hint usage was high, showing that some students used bottom-out hints as worked examples. Worked examples seem valuable for students early on, but hints can provide some of the same scaffolding while encouraging students to self-regulate their learning.

To study the impact of the number of worked examples on learning, We aggregated data across levels 2–6 by high and low track. For the high track, having 0 or 2 worked examples improved solution length; high-track levels with no examples had higher hint usage. These results suggest that the lack of worked examples encouraged some students to choose when to see a hint. High track levels with one example had longer solutions and lower rule application correctness than no examples. Our low track represents true low-proficiency students, and in this track we found that worked examples had a negative impact: increasing hint usage and solution lengths, and decreasing rule accuracy. Together, these results suggest that worked examples detract from learning after Level 1 for both high and low tracks. We also investigated the impact of the order of examples and problem solving. We did not detect any significant differences in time, solution length, or accuracy based on the ordering of worked examples for our high track, and had insufficient data to compare the orderings in the low track. This result is consistent with prior research on worked examples.

References

1. Barnes, T., Stamper, J.: Toward automatic hint generation for logic proof tutoring using historical student data. In: Woolf, B.P., Aïmeur, E., Nkambou, R., Lajoie, S. (eds.) ITS 2008. LNCS, vol. 5091, pp. 373–382. Springer, Heidelberg (2008)
2. Eagle, M.J., Hicks, A.G., Peddycord III, B., Barnes, T.: Exploring networks of problem-solving interactions. In: Learning Analytics and Knowledge (LAK 2015), pp. 21–30 (2015)
3. Kalyuga, S., Chandler, P., Tuovinen, J., Sweller, J.: When problem solving is superior to studying worked examples. J. Educ. Psychol. **93**(3), 579 (2001)
4. McLaren, B.M., Lim, S.-J., Koedinger, K.R.: When and how often should worked examples be given to students? In: Proceedings of the 30th Conference on Cognitive Science Society, pp. 2176–2181. Cognitive Science Society (2008)
5. Mostafavi, B., Liu, Z., Barnes, T.: Data-driven proficiency profiling. In: Educational Data Mining (EDM 2015), pp. 249–252 (2015)

6. Mostafavi, B., Zhou, G., Lynch, C., Chi, M., Barnes, T.: Data-driven worked examples improve retention and completion in a logic tutor. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M.F. (eds.) AIED 2015. LNCS, vol. 9112, pp. 726–729. Springer, Heidelberg (2015)
7. Najar, A., Mitrovic, A.: Should we use examples in intelligent tutors? In: Proceedings of Computers in Education, pp. 5–7 (2012)
8. Trafton, J.G., Reiser, B.J.: Studying examples, solving problems: contributions to skill acquisition (1993)
9. VanLehn, K.: The behavior of tutoring systems. Int. J. Artif. Intell. Educ. **16**(2), 227–265 (2006)