Do Erroneous Examples Improve Learning in Addition to Problem Solving and Worked Examples?

Xingliang Chen(✉), Antonija Mitrovic, and Moffat Mathews

Intelligent Computer Tutoring Group,
University of Canterbury, Christchurch, New Zealand
xingliang.chen@pg.canterbury.ac.nz,
{tanja.mitrovic,moffat.mathews}@canterbury.ac.nz

Abstract. Learning from Problem Solving (PS), Worked Examples (WE) and Erroneous Examples (ErrEx) have all proven to be effective learning strategies. However, there is still no agreement on what kind of assistance (in terms of different learning activities) should be provided to students in Intelligent Tutoring Systems (ITSs) to optimize learning. A previous study [1] found that alternating worked examples and problem solving (AEP) was superior to using just one type of learning tasks. In this paper, we compare AEP to a new instructional strategy which, in addition to PS and WEs, additionally offers erroneous examples to students. The results indicate that erroneous examples prepare students better for problem solving in comparison to worked examples. Explaining and correcting erroneous examples also leads to improved debugging and problem-solving skills.

Keywords: Intelligent tutoring system · Worked examples · Erroneous examples · Assistance · Problem-solving · SQL-Tutor

1 Introduction

A worked example consists of a problem statement, its solution and additional explanations, and therefore provides a high level of assistance to students. WEs reduce the cognitive load on the student's working memory, thus allowing the student to learn faster and deal with more complex problems [2]. Previous research compared the effectiveness of learning from examples to unsupported problem solving [3, 4], and showed that WEs are beneficial for learning in well-structured domains. The benefits of WEs were demonstrated in many studies for novices, but problem solving was found to be superior to WEs for more advanced students [5]. The effects of Problem Solving only (PS), Worked-Examples only (WE), Worked-Examples/Problem-Solving pairs (WE-PS) and Problem-Solving/Worked-examples pairs (PS-WE) have been studied on novices [6]. The WE and WE-PS conditions resulted in significantly higher learning effectiveness compared to the PS and PS-WE conditions. However, van Gog [7] later claimed that the WE-PS and PS-WE conditions were not comparable, because the examples and problems should be identical within and across pairs. Consequently, she

A. Micarelli et al. (Eds.): ITS 2016, LNCS 9684, pp. 13–22, 2016.
DOI: 10.1007/978-3-319-39583-8_2

employed an example-problem sequence (EP condition) and a problem-example sequence (PE condition) for learning. The students learned significantly more in the EP condition than in the PE condition.

In comparison to unsupported problem solving, ITSs provide adaptive feedback, hints and other types of help to students. Several recent studies investigated the effects of learning from WEs compared to learning from tutored problems solving (TPS) in ITSs; a few of those studies found no difference in learning gain but WEs resulted in shorter learning time [8–10]. Contrary to that, a study [1] conducted in SQL-Tutor, a constraint-based tutor that teaches database querying in SQL, found that students learned more from TPS than from WEs; furthermore, the best condition was alternating worked examples with problem solving (AEP), which presented isomorphic pairs of WE and TPS to students.

Several recent studies focused on erroneous examples, which provide incorrect solutions and require students to find and fix errors [11, 12]. Große and Renkl [12] investigated whether both correct and incorrect examples affect learning in the domain of probability. They found that erroneous examples were beneficial on far transfer for high prior knowledge students. Durkin and Rittle-Johnson [11] found that providing both WEs and ErrExs resulted in higher procedural and declarative knowledge in comparison to the WE only condition. They did not find any differences between novices and advanced students.

Surprisingly, there have not been many studies on the benefits of learning from erroneous examples with ITSs. Tsovaltzi et al. [13] investigated the effect of studying erroneous examples of fractions in an ITS. They found that erroneous examples with interactive help improved 6th grade students' metacognitive skills. Furthermore, 9th and 10th graders improved their problem solving skills and conceptual knowledge when using ErrEx with interactive help. Booth et al. [14] demonstrated that students who explained correct and incorrect examples significantly improved their post-test performance in comparison with those who only received WEs in the Algebra I Cognitive Tutor. Additionally, the ErrEx condition and the combined WE/ErrEx condition were beneficial for improving conceptual understanding of algebra, but not for procedural knowledge.

The goal of our study was to investigate the effects of using erroneous examples in addition to WEs and TPS in SQL-Tutor. Previously, the AEP condition was found to be superior to using WEs or TPS alone [1, 15]. In this study, we compared the best condition from that previous study, AEP, to a new instructional strategy which presented a fixed sequence of worked example/problem pairs and erroneous example/problem pairs (WPEP) to support learning. Our hypotheses are that the addition of erroneous examples to WEs and TPS would be beneficial for learning overall (H1), and that their effect would be more pronounced for advanced students (H2).

2 SQL-Tutor

For this study, we modified SQL-Tutor [16], a constraint-based ITS for teaching the Structured Query Language (SQL) by developing three distinct modes to correspond to TPS, WEs and ErrExs. Figure 1 shows the screenshot of the problem-solving interface we used in this study. The left pane shows the structure of the database schema, which the student can explore to gain additional information about tables and their attributes, as well as to see the data stored in the database. The middle pane is the problem-solving environment. At the start of a problem, this pane shows only the input areas for the *Select* and *From* clauses; the student can click on the other clauses to get the input boxes for the remaining clauses as necessary. The right pane shows the feedback once the student submits his/her solution.

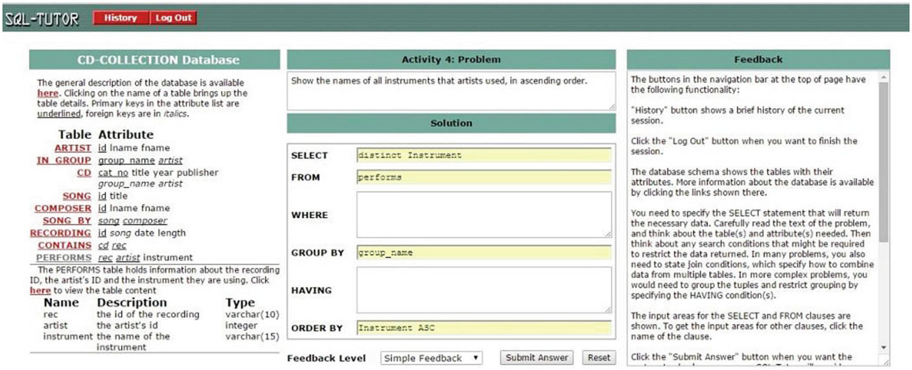

Fig. 1. The student interface of the problem-solving mode of SQL-Tutor

Figure 2 presents the screenshot of the WE mode. An example problem with its solution and explanation is provided in the center pane. A student can confirm that s/he has completed studying the example by clicking the *Continue* button.

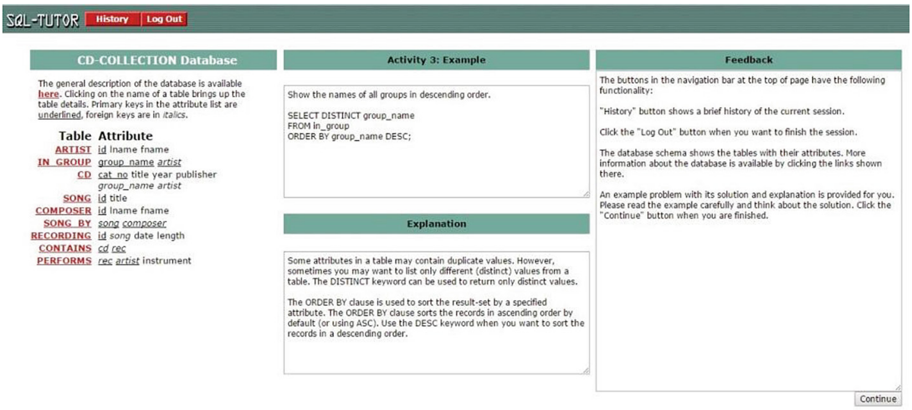

Fig. 2. The student interface of the worked example mode of SQL-Tutor

The ErrEx mode is illustrated in Fig. 3. An incorrect solution is provided for each problem, and the student's task is to analyze the solution, find errors and correct them. The student can submit the solution to be checked by SQL-Tutor multiple times,

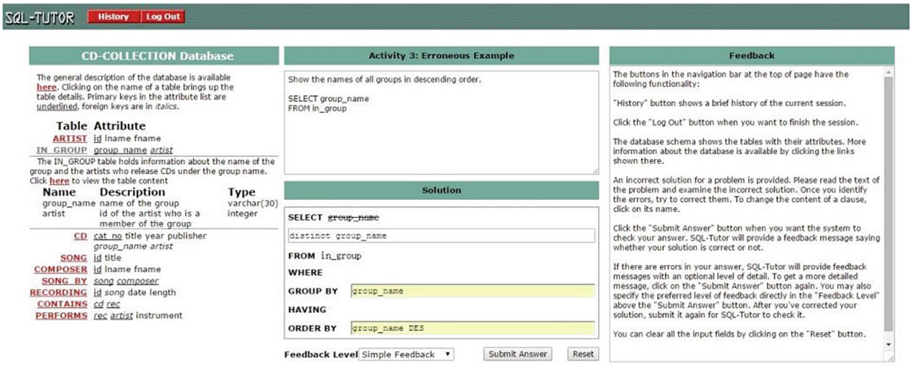

Fig. 3. The student interface of the erroneous-example mode of SQL-Tutor

similar to the problem-solving mode. In the situation illustrated in Fig. 3, the student has identified the SELECT clause as being incorrect, and is defining the new version of it. The student has also added the *Group by* and *Order by* clauses.

Previous research has shown the importance of self-explanation for learning [17, 18]. Providing Self-Explanation (SE) prompts is a common method to encourage students to self-explain. It was found in previous work that WEs help improve conceptual knowledge more than procedural knowledge, whereas problem solving results in higher levels of procedural knowledge [8, 19]. As a consequence, Najar and Mitrovic [1] developed Conceptual-focused Self-Explanation (C-SE) prompts that support students to self-explain relevant domain concepts after problem solving, and Procedural-focused Self-Explanation (P-SE) prompts that supports students to self-explain solution steps after WEs. A C-SE prompt is presented after a problem is solved in order to aid the student in reflecting on the concepts covered in the problem they just completed (e.g. *What does DISTINCT in general do?*). On the other hand, a P-SE prompts are provided after WEs to assist learners in focusing on problem-solving approaches (e.g. *How can you specify a string constant?*). C-SE and P-SE prompts were used in the previous study [1] to increase learning. In order to keep our experimental design consistent with that of [1], our participants received C-SE prompts after problems, and P-SE prompts after WEs, to complement learning activities so that both conceptual and procedural knowledge is supported. Since ErrExs contain both properties of problems and WEs, we provided P-SE and C-SE prompts alternatively after ErrExs.

3 Experimental Design

The study was conducted with 60 students enrolled in an introductory database course at the University of Canterbury, in regular labs scheduled for the course (100 min long). Prior to the study, the students learned about SQL in lectures, and had one lab session. The version of SQL-Tutor used in this study had two conditions: Alternating Examples and Problems (AEP), the most effective learning condition from the previous study [15], and the experimental condition consisting of Worked example/Problem pairs and Erroneous example/Problem pairs (WPEP). In both conditions, the order of tasks was the same, with the only difference being whether tasks were presented as problems to be solved, WEs or ErrExs. After providing informed consent, the participants were randomly assigned to either AEP or WPEP. The pre-test was administered online, followed by the 20 learning tasks. After completing all tasks, the participants completed the online post-test, which was similar in complexity and length to the pre-test. Figure 4 illustrates the study design.

AEP	**WPEP**
Pre-test	
20 problems and WEs (10 isomorphic pairs)	10 problems/WEs (5 isomorphic pairs), and 10 problems/ErrEx (5 isomorphic pairs), presented in alternation
Post-test	

Fig. 4. Study design with two conditions (AEP and WPEP)

4 Results

Our study was conducted at a time when the participants had assessment due in other courses they were taking. Since participation was voluntary, not all participants completed the study. Twenty-six students completed all activities and the post-test. In the following section, we present the results of analyses performed on the data collected for those 26 students (15 in the AEP and 11 in the WPEP condition).

More than half of the participants have not completed the study. Such a big attrition rate necessitated a further investigation. We compared the incoming knowledge (i.e. the pre-test scores) of the participants who completed or abandoned the study, in order to identify whether they were comparable or whether it was the weaker students who have not completed the study.

The pre/post-test consisted of 11 questions each. Questions 1–6 measured conceptual knowledge and were multi-choice or true-false questions (with the maximum of 6 marks). Questions 7–9 focused on procedural knowledge; question 7 was a multi-choice question (one mark), followed by a true-false question (one mark), while question 9 required the student to write a query for a given problem (4 marks). The last two questions presented incorrect solutions to two problems, and required the student to correct them, thus measuring debugging knowledge (6 marks). Therefore, the maximum mark on each test was 18.

The pre-test scores are given in Table 1. There were no significant differences between the two subsets of participants on overall pre-test scores. There were also no significant differences on the scores for declarative, procedural and debugging questions. Therefore, the 26 remaining participants are representative of the class.

Table 1. Pre-test scores (in %) for all students, and for participants who completed/abandoned the study (standard deviations shown in parentheses)

	All participants (60)	Completed (26)	Abandoned (34)
Overall	65.14 (14.09)	65.81 (13.14)	64.62 (14.96)
Conceptual	55.28 (17.76)	53.85 (17.19)	56.37 (18.36)
Procedural	81.67 (23.26)	85.26 (16.72)	78.92 (27.16)
Debugging	58.47 (23.19)	58.33 (24.15)	58.58 (22.79)

4.1 Do the Conditions Differ on Learning Outcomes?

We used the Mann-Whitney U test to analyze the differences between the two conditions (Table 2). There was no significant difference between AEP and WPEP in both the pre-test and post-test scores. The students in both the AEP ($p = .001$) and the WPEP condition ($p = .003$) improved significantly between pre-test and post-test, as confirmed by a statistically significant median increase identified by the Wilcoxon singed-rank test (shown in the *Improvement* row of Table 2). The effect sizes (Cohen's d) are high for both groups, with the WPEP group having a higher effect size. For both groups, the pre-test and post-test scores are positively correlated, but only the correlation for AEP is significant.

Table 2. Basic statistics for the two conditions

	AEP (15)	WPEP (11)
Pre-test (%)	67.22 (15.37), med = 66.67	63.89 (9.7), med = 61.11
Post-test (%)	91.11 (12.92), med = 97.22	93.94 (6.67), med = 94.44
Improvement	$W = 120$, $p < .005$, $d = 1.29$	$W = 66$, $p < .005$, $d = 1.73$
Pre/post-test correlation	$r = .58$, $p < .05$	$r = .52$, ns
Interaction time (min)	65.64 (16.96)	67.09 (10.22)

On average, the participants spent 66 min interacting with the learning tasks. There was no significant difference on the total interaction time between the two conditions. The students in both groups solved the same number of problems (10). The AEP group had 10 WEs, while the WPEP group had five WEs and five ErrExs. We expected erroneous examples to take more time in comparison to WEs, but the difference was not significant.

Table 3. Detailed scores on pre/post-tests

Group	Questions	Pre-test %	Post-test %	W, p
AEP (15)	Conceptual	57.78 (17.67)	94.44 (10.29)	120, .001**
	Procedural	80.56 (18.28)	97.78 (5.86)	36, .011**
	Debugging	63.33 (24.56)	81.11 (29.46)	73, .054*
WPEP (11)	Conceptual	48.48 (15.73)	91 (8.7)	66, .002**
	Procedural	91.67 (12.36)	97.73 (7.54)	ns
	Debugging	51.51 (22.92)	93.18 (15.28)	45, .007**

Table 3 shows the scores on different question types. There were no significant differences on pre-test scores for the two conditions. In the AEP condition, there were significant differences between pre- and post-test scores on conceptual and procedural questions, as well as a marginally significant difference on the score for debugging questions. In the WPEP condition, the students' scores on conceptual and debugging questions increased significantly between pre- and post-test, but there was no significant difference on the scores on procedural questions. The WPEP group started with a very high level of procedural knowledge, and that explains no significant difference on this type of questions.

In order to identify whether the two conditions affected students' problem solving differently, we analyzed the log data. As explained previously, ten learning tasks were problems to be solved. Table 4 reports the number of attempts (i.e. solution submission), as well as the number of errors (i.e. the number of violated constraints) for the ten problems. Overall, the AEP group made significantly more attempts (U = 37.5, p = .018) and more mistakes (U = 44, p = .047) on the ten problems.

Table 4. Analysis of attempts and errors for the two conditions

	All problems		Problems 4, 8, 12, 16, 20		Problems after WEs	
	Attempts	Errors	Attempts	Errors	Attempts	Errors
AEP	4.54 (1.7)	12.87 (8.31)	5.67 (2.14)	17.44 (11.12)	3.41 (1.89)	8.29 (8.09)
WPEP	3.08 (1.06)	7.73 (6.75)	3.49 (1.43)	9.64 (10.47)	2.67 (1.21)	5.82 (7.1)
p	<.02**	<.05**	<.01**	<.05**	ns	ns

The table also reports the two measures for various subsets of problems, identified on the basis of the previous learning task. We wanted to investigate whether correct and erroneous examples prepare students differently for problem solving. Problems 4, 8, 12, 16 and 20 were presented in the WPEP condition after ErrEx, whereas in the AEP condition after WEs. For those five problems, there were significant differences between the two conditions on both attempts (U = 30, p = .005) and errors (U = 41, p = .032). On the other hand, problems 2, 6, 10, 14 and 18 were presented to both conditions after WEs. For those problems, we found no significant differences between the two groups on either attempts or errors on this subset of problems. These findings show that erroneous examples prepare students better for problem solving in comparison to worked examples, which confirms our hypothesis H1. This is important, as some of the previous studies (as discussed in the Introduction) have found that worked examples are superior to other types of learning tasks.

4.2 Comparing Novices and Advanced Students

We were also interested in the effectiveness of the two conditions on students with different levels of pre-existing knowledge. We classified students into novices and advanced students based on their pre-test scores (Table 5). The participants whose pre-test scores are lower than 66 % (the overall median pre-test score for our sample) are considered to be novices, and the rest as advanced students.

The Mann-Whitney U test revealed no significant differences between novices/advanced students in the two conditions, on pre- and post-test scores. The Wilcoxon signed-rank test showed that novices and advanced students in both conditions improved significantly between the pre- and post-test ($p < 0.05$). A deeper analysis of the pre/post-test scores revealed that in the WPEP condition, the score for debugging questions improved significantly for novices ($p < .05$) and marginally significantly for advanced students ($p = .059$), while only advanced students from the AEP condition improved their score on debugging questions ($p = .01$). The novice AEP students did not improve their debugging knowledge. The normalized gain on debugging questions only for novices from the AEP condition was 0.15 (sd = 0.71), while for novices from the WPEP group it was 0.76 (0.3); the difference is marginally significant ($U = 29.5$, $p = .063$, $d = 0.96$). The fact that both advanced and novice WPEP students improved on debugging questions rejects our second hypothesis; contrary to our expectations, both novices and advanced students benefitted from ErrEx.

Table 5. Comparing novices and advanced students

		Score (%)	Pre-test	Post-test	W, p
AEP (15)	Novices (6)	Overall	52.31 (7.94)	80.09 (13.77)	21, .028**
		Debug. questions	41.67 (20.41)	56.94 (34.73)	ns
	Adv. (9)	Overall	77.16 (9.8)	98.46 (3.7)	45, .008**
		Debug. questions	77.78 (14.43)	97.22 (5.89)	36, .01**
WPEP (11)	Novices (6)	Overall	56.94 (3.4)	91.2 (7.54)	21, .028**
		Debug. questions	38.89 (13.61)	87.5 (19.54)	15, .043**
	Adv. (5)	Overall	72.24 (7.85)	97.22 (3.93)	15, .041**
		Debug. questions	66.67 (23.57)	100 (0)	10, .059*

5 Discussion and Conclusions

Previous studies show that WEs are beneficial for novices in comparison to problem solving [6, 15, 20]. In a previous study, alternating WEs with problem solving was found to the best strategy in SQL-Tutor [1]. However, the inclusion of ErrEx has not been studied before in this instructional domain. In this study, we compared students' performance in two conditions: alternating worked examples/problem (AEP), and worked example/problem pairs and erroneous examples/problem pairs (WPEP).

We found no significant difference between AEP and WPEP conditions on pre- and post-test performance, but the participants in both conditions improved significantly their scores on the post-test from the pre-test. Students in the WPEP condition acquired more debugging knowledge than those in the AEP condition. A possible explanation is that extra learning and additional time in the correcting phase of erroneous examples contribute to this benefit. Furthermore, students who learned with erroneous examples showed higher performance on problem solving as measured by the number of attempts per problems and also the number of mistakes made. This suggests that the erroneous examples aid learning more than worked examples, which confirmed our hypothesis H1. The WPEP participants learned from both worked examples and erroneous examples. When students were asked to identify and correct errors in ErrEx, they engaged in deeper cognitive processing in comparison to when they engage with WEs. Therefore, they were better prepared for concepts required in the next isomorphic problem compared to the situation when they received WEs.

Although the present results suggest that ErrExs aid learning, an important issue concerns the benefit for students with different knowledge levels. Hypothesis H2, like in [12], was that advanced students would learn more from erroneous examples than novices. However, we did not find a difference between novices and advanced students in WPEP; both subgroups improved their debugging knowledge. Furthermore, novices from the WPEP group improved their debugging knowledge significantly more than their peers of similar abilities from the AEP group (with the effect size close to 1 sigma). Therefore, the students with any knowledge level benefitted from erroneous examples. One of the possible explanations for a different finding in comparison to [12] is in the instructional domains used in each study. The instructional task of the Große and Renkl study was probability (a well-defined instructional task), while the students were specifying SQL queries for ill-defined tasks in our study.

One of the limitations of our study is the small sample size. The timing of the study coincided with assignments in other courses the participants were taking, so many participants did not complete the full study. We plan to conduct the same study with a larger population. McLaren et al. [21] found that erroneous examples led to a delayed learning effect. However, our study did not include a delayed test. It would be interesting to see the results of the delayed learning effect.

Our study demonstrated that an improved instructional strategy, WPEP, resulted in improved problem solving, and that it also benefitted students with various levels of prior knowledge in SQL-Tutor. The results suggest that the students with different levels of prior knowledge may perform differently with worked examples, erroneous examples, and problem-solving. In addition, all students in our study learned SQL in the lectures before participating in our study. One direction for future work would be to develop an adaptive strategy that decides what learning activities (TPS, WE or ErrEx) to provide to the student based on his/her student model.

References

1. Shareghi Najar, A., Mitrovic, A.: Examples and tutored problems: how can self-explanation make a difference to learning? In: Lane, H., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS, vol. 7926, pp. 339–348. Springer, Heidelberg (2013)
2. Sweller, J., Van Merrienboer, J.J., Paas, F.G.: Cognitive architecture and instructional design. Educ. Psychol. Rev. **10**(3), 251–296 (1998)
3. van Gog, T., Rummel, N.: Example-based learning: integrating cognitive and social-cognitive research perspectives. Educ. Psychol. Rev. **22**(2), 155–174 (2010)

4. Atkinson, R.K., Derry, S.J., Renkl, A., Wortham, D.: Learning from examples: instructional principles from the worked examples research. Rev. Educ. Res. **70**(2), 181–214 (2000)
5. Kalyuga, S., Chandler, P., Tuovinen, J., Sweller, J.: When problem solving is superior to studying worked examples. Educ. Psychol. **93**(3), 579 (2001)
6. van Gog, T., Kester, L., Paas, F.: Effects of worked examples, example-problem, and problem-example pairs on novices' learning. Contemp. Educ. Psychol. **36**(3), 212–218 (2011)
7. van Gog, T.: Effects of identical example-problem and problem-example pairs on learning. Comput. Educ. **57**(2), 1775–1779 (2011)
8. Schwonke, R., Renkl, A., Krieg, C., Wittwer, J., Aleven, V., Salden, R.: The worked-example effect: not an artefact of lousy control conditions. Comput. Hum. Behav. **25**(2), 258–266 (2009)
9. McLaren, B.M., Isotani, S.: When is it best to learn with all worked examples? In: Biswas, G., Bull, S., Kay, J., Mitrovic, A. (eds.) AIED 2011. LNCS, vol. 6738, pp. 222–229. Springer, Heidelberg (2011)
10. McLaren, B.M., Lim, S.-J., Koedinger, K.R.: When and how often should worked examples be given to students? New results and a summary of the current state of research. In: Proceedings of 30th Annual Conference of the Cognitive Science Society, pp. 2176–2181 (2008)
11. Durkin, K., Rittle-Johnson, B.: The effectiveness of using incorrect examples to support learning about decimal magnitude. Learn. Instr. **22**(3), 206–214 (2012)
12. Große, C.S., Renkl, A.: Finding and fixing errors in worked examples: can this foster learning outcomes? Learn. Instr. **17**(6), 612–634 (2007)
13. Tsovaltzi, D., McLaren, B.M., Melis, E., Meyer, A.-K.: Erroneous examples: effects on learning fractions in a web-based setting. Technol. Enhanc. Learn. **4**(3–4), 191–230 (2012)
14. Booth, J.L., Lange, K.E., Koedinger, K.R., Newton, K.J.: Using example problems to improve student learning in algebra: differentiating between correct and incorrect examples. Learn. Instr. **25**, 24–34 (2013)
15. Najar, A.S., Mitrovic, A.: Do novices and advanced students benefit differently from worked examples and ITS? In: Proceedings of International Conference Computers in Education, pp. 20–29 (2013)
16. Mitrovic, A.: An intelligent SQL-Tutor on the web. Artif. Intell. Educ. **13**, 173–197 (2003)
17. Weerasinghe, A., Mitrović, A.: Studying human tutors to facilitate self-explanation. In: Ikeda, M., Ashley, K.D., Chan, T.-W. (eds.) ITS 2006. LNCS, vol. 4053, pp. 713–715. Springer, Heidelberg (2006)
18. Chi, M.T., Leeuw, N., Chiu, M.H., LaVancher, C.: Eliciting self-explanations improves understanding. Cogn. Sci. **18**(3), 439–477 (1994)
19. Kim, R.S., Weitz, R., Heffernan, N.T., Krach, N.: Tutored problem solving vs."pure" worked examples. In: Proceedings of 31st Annual Conference of the Cognitive Science Society. Cognitive Science Society (2009)
20. McLaren, B.M., van Gog, T., Ganoe, C., Yaron, D., Karabinos, M.: Exploring the assistance dilemma: comparing instructional support in examples and problems. In: Trausan-Matu, S., Boyer, K.E., Crosby, M., Panourgia, K. (eds.) ITS 2014. LNCS, vol. 8474, pp. 354–361. Springer, Heidelberg (2014)
21. McLaren, B.M., Adams, D.M., Mayer, R.E.: Delayed learning effects with erroneous examples: a study of learning decimals with a web-based tutor. Artif. Intell. Educ. **25**(4), 520–542 (2015)

http://www.springer.com/978-3-319-39582-1

Intelligent Tutoring Systems
13th International Conference, ITS 2016, Zagreb,
Croatia, June 7-10, 2016. Proceedings
Micarelli, A.; Stamper, J.; Panourgia, K. (Eds.)
2016, XXI, 552 p. 100 illus., Softcover
ISBN: 978-3-319-39582-1