How Do Learners Behave in Help-Seeking When Given a Choice?

Sebastian Gross and Niels Pinkwart

Humboldt-Universität zu Berlin, Germany
{sebastian.gross,niels.pinkwart}@hu-berlin.de

Abstract. We describe the results of a study that investigated learners' help-seeking behavior using two feedback options implemented in an ITS for Java programming. The 25 students had the choice between asking for feedback on errors in their programs and feedback on possible next steps in the solution process. We hypothesized that learners' choices would depend on correctness of their programs and their progress in problem-solving. Surprisingly, this hypothesis was not confirmed.

Keywords: intelligent tutoring system, help-seeking, feedback choice

1 Introduction

Many ITS systems favor a feedback on demand model, as this supports self-regulated learning better than feedback on the initiative of the system and avoids risks of undesired interference when feedback messages disturb learners' cognitive processes. Yet, if feedback is presented upon request, the help-seeking behavior of the learner plays an important role for the success of the learning technology. Previous research has investigated aspects of learners' help-seeking attitudes, including detection of misuse [2] and instructing learners to improve their help-seeking behavior [1].

An under-investigated question in this area is how learners would behave when given a choice between two or more help options: are the learners' choices in line with what the feedback options are intended to be used for? We previously [4] proposed a novel approach for providing feedback in ITSs by employing example-based learning. Research on worked examples that provide an expert's solution on how to solve a given problem has proven to be effective in various learning domains such as mathematics [5] or programming education [3]. In our approach, prototype-based classification of dissimilarity data is used to identify an appropriate example from a data set consisting of successful and unsuccessful learners' solution attempts and sample solutions (or parts thereof) created by experts. This selected element of the data set (called counterpart) is then used to provide feedback to a learner by presenting the learner's current solution attempt, contrasting it with the selected element of the data set, and asking her to compare the two. In previous work [4], we implemented the approach in an ITS for Java programming and tested it with students of an introductory programming course, comparing four strategies (randomly chosen by the ITS) for selecting an appropriate example from a data set. The results supported the hypothesis that using a data set consisting of expert solution steps is superior to using complete sample solutions only and to using learner solution attempts only. The previous study also suggested that two feedback strategies seem to be most promising: selecting the most similar sample solution part in the data set, or selecting the next step of this most similar solution. However, it remained unclear *when* (depending on learners' progress) the most similar sample solution step or its next step should be provided as feedback to help learners fix mistakes or proceed in problem-solving. This is the question investigated in this paper.

2 Study Description

In order to answer the question stated above, we modified our ITS for Java programming in such a way that the feedback strategy is not randomly chosen by the system anymore. Instead, we allowed learners to choose between two feedback options. In both options, learners' current programs were analyzed and compared to the data set consisting of sample solution parts created by experts. The difference between the feedback options was that in feedback option 1 (marked with *"I don't know where the error in my program is."*), the most similar sample solution step in the data set was selected as counterpart. In feedback option 2 (marked with *"I don't know how to continue with my program."*), the next step of the most similar sample solution part was selected as counterpart. Based on the counterpart, the system then provided feedback to learners. In accompanying text, the system clearly informed learners about the difference between the two options, suggesting that option 1 would likely be more appropriate if students did not know how to fix an error in their program (since the feedback shows a similar but correct part of a sample solution), and option 2 would be preferable if students did not know how to proceed (since the feedback shows a more advanced part of a sample solution).

With this study design, we wanted to investigate the help-seeking behavior of students in this example based feedback provision scenario: do students select the feedback option that is (probably) more helpful for them? Specifically, our hypotheses were that:

- H1 if their current program is erroneous, learners would ask for the most similar sample solution step, and
- H2 if their current program is correct (but not necessarily complete), learners would ask for the next step of the most similar sample solution step

To evaluate the hypotheses, we conducted a field study in which the ITS was used in the context of an introductory programming course at Humboldt-Universität zu Berlin. The system was shown to the students in class, and the content available in the system (exercises, sample solution steps etc.) matched the course content. Students had the opportunity to use the system over a period of 10 weeks from anywhere they wanted at anytime. Participation was completely voluntary and possible with a self-chosen login.

3 Results

For 39 programming tasks, 25 students requested feedback 340 times. They chose option 1 in 187 cases and option 2 in 153 cases. We asked an experienced Java tutor to assess each student program as to whether (i) the program is syntactically correct, and (ii) the program is on target with respect to the given problem. For the latter criterion, we measured the interrater reliability by having a 10% random sample assessed by a second experienced Java tutor, resulting in an acceptable Cohen's kappa of $\kappa = .58$. We also asked the human tutor to identify misuse of the feedback options where students requested multiple feedbacks within a few minutes without (substantially) changing their programs. Overall the expert identified 116 cases of such misuse. As stated in Section 2, we

<table>
<tr><th></th><th colspan="2">Feedback option 1
(show similar sample)</th><th colspan="2">Feedback option 2
(show next step)</th><th>Total</th></tr>
<tr><td>syntax error but program on target</td><td>32 (52)</td><td rowspan="2">55% (52%)</td><td>15 (24)</td><td rowspan="2">52% (46%)</td><td>47 (76)</td></tr>
<tr><td>syntax error and program not on target</td><td>38 (44)</td><td>35 (46)</td><td>73 (90)</td></tr>
<tr><td>syntax correct and program on target</td><td>48 (75)</td><td rowspan="2">45% (48%)</td><td>21 (28)</td><td rowspan="2">48% (54%)</td><td>69 (103)</td></tr>
<tr><td>syntax correct but program not on target</td><td>9 (16)</td><td>26 (55)</td><td>35 (71)</td></tr>
<tr><td>Total</td><td colspan="2">127 (187)</td><td colspan="2">97 (153)</td><td>224 (340)</td></tr>
</table>

Table 1. Program states and chosen feedback options. Numbers in parentheses include the feedback requests classified as misuse. Choices that we expected to be chosen by a learner (depending on her program's state) are shaded in gray.

expected that feedback option 1 would mainly be chosen by students whose programs are erroneous (i.e., have syntax errors or are not on target) while option 2 would mainly be chosen by students who got stuck in problem-solving (but their programs are syntactically correct and on target). Table 1 summarizes the observed student behavior, numbers in parentheses include the feedback requests classified as misuse. When feedback option 1 was chosen, only 55% (52%) of the programs were syntactically erroneous, while only 48% (54%) of the programs were syntactically correct when feedback option 2 was chosen. It is obvious that this data does not support the hypotheses. If we consider whether a student's program was on target as an indicator of correctness, the data even shows a picture completely contrary to our expectations: if feedback option 1 was chosen, the correct student's program was on target in 63% (68%) of the cases, whereas this was true only in 37% (34%) of the cases if feedback option 2 was chosen. This difference is statistically significant ($\chi^2 = 14.75$ (38.85)$, df = 1, p < .001$). While Table 1 contains all feedback requests regardless of individual student behavior, we were also interested in individual student characteristics: were there some students with help-seeking behavior as hypothesized and others who behaved differently, or did all students exhibit a more or less homogeneous help-seeking attitude? We therefore examined how often a student chose feedback option 1

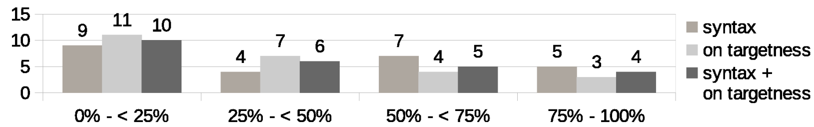

Fig. 1. Percentages of how often students chose feedback options as hypothesized

when her program was syntactically erroneous or not on target, and how often a student chose feedback option 2 when her program was syntactically correct and on target. Figure 1 shows the results, grouped into four ranges where each range indicates in how many cases a learner behaved as hypothesized in Section 2. The diagram illustrates that learners' help-seeking behavior cannot be distinctly divided into "predicted" and "unpredicted" but the number of learners is almost evenly distributed in the four ranges. This implies that indeed for some learners our hypotheses hold, but not for others.

4 Conclusion and Future Work

We investigated learners' help seeking behavior when given a choice, hypothesizing that learners' choice would depend on the correctness of their programs. Yet, the hypotheses were not confirmed in the study - to the contrary: when considering on-targetness, learners' choices were largely in contrast with our prediction. There are some possible explanations for this: (1) learners may be fully aware of what kind of help they actually need, but the underlying factors of this reasoning do not correlate with our predictions, (2) the hypotheses indeed hold but learners are not aware of what kind of help they actually need (i.e., they do not know if they have a problem to fix or not before they can make the next step), and they probably need help-seeking assistance, or (3) the hypotheses hold only for some learners (as suggested by Figure 1) and we need to determine the factors that account for this. In future work, we plan to address these issues by conducting a qualitative analysis of students' programs before and after requesting feedback.

References

[1] V. Aleven, B. Mclaren, I. Roll, and K. Koedinger. Toward meta-cognitive tutoring: A model of help seeking with a cognitive tutor. *IJAIED*, 16(2):101–128, 2006.
[2] R. S. Baker, A. T. Corbett, and K. R. Koedinger. Detecting student misuse of intelligent tutoring systems. In *ITS*, pages 531–540. Springer, 2004.
[3] P. Brusilovsky and M. Yudelson. From webex to navex: Interactive access to annotated program examples. *Proceedings of the IEEE*, 96(6):990–999, june 2008.
[4] S. Gross, B. Mokbel, B. Hammer, and N. Pinkwart. How to select an example? a comparison of selection strategies in example-based learning. In S. Trausan-Matu, K. E. Boyer, and K. Crosby, M.and Panourgia, editors, *Proc. of ITS*, pages 340–347, Berlin, Germany, 2014. Springer Verlag.
[5] A. Renkl. Learning mathematics from worked-out examples: Analyzing and fostering self-explanations. *Europ. Journ. of Psych. of Education*, 14(4):477–488, 1999.