Trying to Reduce Bottom-out hinting: Will telling student how many hints they have left help?

Yu Guo, Joseph E. Beck, Neil T.Heffernan

Wocester Polytechnic Insititute
Department of Computer Science
100 Institute Road
Worcester, MA 01609
{yuguo,nth}@wpi.edu
joseph.beck@gmail.com
phone:508-831-5006

Abstract. Many model-tracing intelligent tutoring systems give, upon demand, a series of hints until they reach the bottom-out hint that tells them exactly what to do (exact answer of the question). Since students don't know the number of hints available for a given question, some students might be surprised to, all of a sudden, be told the final answer; letting them know when the bottom out hint is getting close should help cut down on the incidence of bottom-out hinting. We were interested in creating an intervention that would reduce the chance that a student would ask for the bottom out hint. Our intervention was straightforward; we simply told them the number of hints they had not yet seen so that they could see they were getting close to the bottom out hint. We conducted a randomized controlled experiment where we randomly assigned classrooms to conditions. Contrary to what we expected, our intervention led to more, not less, bottom out hinting. We conclude that the many intelligent tutoring systems that give hints in this manner should not consider this intuitively appealing idea.

1 Introduction

Providing help to students when they get stuck in the learning process is a pedagogical strategy in Intelligent Tutoring System (ITS). Showing student hints is a popular way of guiding students in most tutoring system. The Andes Physics Tutoring System gives learners different levels of hints on demand in their process of learning [1]. Cognitive tutor also gives hints on demand [2].Our system ASSISTment also provides, for each question, some unspecified number of hints, which eventually will terminate with the student being told exactly what to do. However, none of them informs students when they will get the bottom-out hint. Our hypothesis is that students might not want to be told the exact answer, but in the current systems they can never be sure if the next time they ask for a hint, they will get told the answer. We want the student to think over the problem before he goes through all the hints to the bottom one. In other words, the intelligent tutoring system needs to prevent the student from "gaming the system" [3] by cheating the system to reach a correct answer. Others have also investigated ways to reduce students' gaming behavior ([4, 5]).The goal of this research is to find if telling students how many hints there are left in the problem will lead students to less often reach bottom-out hints.

We carried out the experiment on ASSISTment, a web-based tutoring system developed by Worcester Polytechnic Institute. It assists student learning math by providing individual help in the manner of giving a sequence of hints and scaffolding questions while assessing student. Teachers are able to keep track of students' performance in the system through several up-to-date reports generated by the system. Content builders are provided the tool to build main questions, scaffolding questions, hints, answers, buggy messages and so on so forth in a convenient way. The system is freely available at www.assistment.org. ASSISTment is now used by thousands of kids and teachers as a normal part of their classes. We changed the system's help request interface for this experiment and built a logistic regression model to analyze the data.

2. Experiment

Our approach is as following. Instead of showing the button labeled with "Request Help," we count how many hints are left in the problem for a specific student and label the help button as "Show me hint <N1> out of <N2>". N2 equals total hints of the problem and N1 means how many hints the student has asked if he clicks on the button. When student is about to reach the bottom hint, we will label the button as "Show me the last hint", explicitly telling them this hint will show the exact answer of the question. We collected data from 2007/9/30 to 2007/11/29. Roughly halfway through this time period, on 10/29, we turned this intervention "on" in half the classrooms, leaving the other classrooms to act as a control to control for any existing trends over time in student behavior. For each teacher in our system, we randomly selected half of the classrooms to be in the experiment, while the other classes served as the control.

We had 320 students that did more than 20 problems both before and after 10/29 with 88 students in experiment group and 232 students in control group. We filtered the data by 20 problems for the reason that we wanted to focus on the behavior of consistent users of ASSISTment during the time period.

The data was collected on question level. Each row includes the information about how one specific student performs on one particular problem. We collected the information of "student_id", "hit_bottom" (whether the student reaches bottom-out hint in this problem), "easiness" (easiness of problem which is measured by correctness over all the students in the system), condition (whether the student in experiment group or control group), "beforeEx" (whether this problem is done pre-intervention or post-intervention). The reason we collected data in this manner is that in our expectation, whether the student will hit the bottom-out hint relates to the problem easiness level, student himself, which condition the student is in and when this problem is done. There are more than 37,000 rows in our dataset.

3. Result and Analysis

We built a logistic regression model to predict dependent variable "hit_bottom" with factors "student_id", "condition", "beforeEx" and covariate "easiness". Since these are elements affecting students' bottom-out hinting behavior, we consider them as main effects in our model and moreover, we take beforeEx * condition into account as an interaction effect.

The parameter estimates of the model are shown in Table 1. The higher the β value is, the less likely the student will hit the bottom-out hint. The fact that the β value for control group is 0.280 and with P=0.547 suggests that there is no significant change on students hinting behavior whether he is in control group or experiment group when ignoring the time period. The fact the β value for "After Intervention" is -0.092 with P =0.151 tells us that students hit more bottom-out hints after the intervention. However, this is not statistically reliable. We can not get conclusion related to the effect of intervention by simply looking at these two parameters, which is the reason that we take the interaction of these two parameters into consideration. The Easiness parameter of +5.441 suggests that there is more bottom out hinting on harder items, which is to be expected. Finally, and most interesting, the interaction between beforeEx*condition, is the crucial parameter that allows us to see if there is a change in student behavior due to the experiment, after taking into account all the other parameters in the model. The fact the parameter for [Condition=C]*[beforeEx=0] is 0.310 indicates that the experiment caused more bottom out hinting, contrary to our hypothesis. This difference was statistically reliable.

Table 1. Parameter Estimates

	β	Sig.
Control Group(Condition=C)	0.280	0.547
Experiment(Condition=E)	0*	-
After Intervention(beforeEx=0)	-0.092	0.151
Before Intervention(beforeEx=1)	0*	-
Control Group and After Intervention([Condition=C]*[beforeEx=0])	0.310	<0.0002
Control Group and Before Intervention([Condition=C]*[beforeEx=1])	0*	-
Experiment Group and After Intervention([Condition=E]*[beforeEx=0])	0*	-
Control Group and After Intervention([Condition=E]*[beforeEx=1])	0*	-
Easiness	5.441	<0.00001

*This parameter is set to zero because it is redundant

For further analysis, we divided the students into three groups with high, medium and low math proficiency according to the student proficiency parameter estimated by the one-parameter Item Response Theory model (Rasch model)[1]. The Rasch model was trained based on data collected in ASSISTment system from Sept. 2004 to Jan. 2008 that includes 14273 students' responses to 2796 main questions [6]. Since the intervention might have different effect on distinct population, the analysis in the view of student's proficiency came next.

We built a logistic regression model for each group. We will put our main focus on the interaction effect. In Table2, we show the β values for interaction effect in each group. The β value of interaction is 0.274 with P=0.013 indicates that for the population with low math proficiency, the intervention led to more bottom-out hints. It is the same case as population with medium math proficiency, giving β value equals 0.477 and P value less than 0.002. Moreover, the effects are statistically reliable. However, for population with high math proficiency, the intervention has no statistically reliable effect.

Table 2. Parameter Estimates for three groups

[Condition=C]*[beforeEx=0]	β	Sig.
Low proficiency group	0.274	0.013
Medium proficiency group	0.477	<0.002
High proficiency group	0.061	0.777

4. Conclusion and Future work

This experiment shows that telling student how many hints they have left affects student behavior regarding the bottom-out hint. The logistic regression shows that the change will in general lead to **more** bottom-out hints, contrary to our hypothesis. Further analysis in terms of different population indicates that the intervention will lead to **more** bottom-out hints in the population with low and medium math proficiency while has no statistically reliable effect on the students with high proficiency. According to the results, telling students how many hints left is not a suggestion that other tutoring systems designers should implement. One explanation of this result is that students might not believe that asking for hints, or getting the bottom out hint, hurts their learning. For those students who are aimed at "gaming the system," showing the number of remaining hints might assist or encourage their gaming behavior.

We will continue working on how to prevent students from asking for more help than they probably need in the learning process. As it has been stated that students might not consider that asking for more help than they need hurts their learning, an intervention as following seems to be a potential strategy. For instance, we can provide a partial credit for each question with some initial value and show it to the student. Whenever student asks for help, we will take some points off the credit. Hopefully, this intervention would to some extent guide students to contribute more

[1] We won't go further into "Item Response Theory model" in this paper. Basically, it gives us a statistically reliable estimate of student's math proficiency based on their entire responses in the interaction with ASSISTment.

efforts to get the question done correctly by themselves or based on the help they've already attained from the system. Moreover, we will focus on analyzing if students gain more learning through the intervention, which is an important metric of intelligent tutoring systems.

5. Acknowledgement

We would like to thank Worcester Public Schools and all of the people associated with developing the ASSISTment system listed at www.ASSISTment.org who made this experiment possible. We would also like to acknowledge funding from the US Department of Education, the National Science Foundation, the Office of Naval Research and the Spencer Foundation. All of the opinions expressed in this paper are those solely of the authors and not those of our funding organizations.

References

1. VanLehn, K., Lynch, C., Schulze, K., Shapiro, J.A., Shelby, R. Taylor, L., Treacy, D., Weinstein, A. & Wintersgill, M. (2005). The Andes Physics Tutoring System: Lessons Learned. *International Journal of Artificial Intelligence in Education*, 15(3), p.1-47.
2. Koedinger, K. R., Anderson, J. R., Hadley, W. H. & Mark, M. (1997). Intelligent tutoring goes to school in the big city. *International Journal of Artificial Intelligence in Education*, 8, p.30-43
3. Baker, R. (2007). Is Gaming the System State-or-Trait? Educational Data Mining Through the Multi-Contextual Application of a Validated Behavioral Model. *Workshop on Data Mining for User Modeling, Educational Data Mining Track, at User Modeling*, p. 76-80.
4. Walonoski, J. & Heffernan, N.T. (2006a). Detection and analysis of off-task gaming behavior in intelligent tutoring systems. In Ikeda, Ashley & Chan (Eds.). *Proceedings of the Eight International Conference on Intelligent Tutoring Systems*. p. 382-391.
5. Lloyd, N., Heffernan, N.T. & Ruiz, C. (2007). Predicting student engagement in intelligent tutoring systems using teacher expert knowledge. In the *Educational Data Mining Workshop* held at the 13th Conference on Artificial Intelligence in Education. P.40-49
6. Feng, M., Beck, J., Heffernan, N.T. & Koedinger, K.R. (Submitted). Can an Intelligent Tutoring System Predict Math Proficiency as Well as a Standardized Test? Paper submitted to the 1st International Annual Conference on Education Data Mining. Montreal 2008. (DRAFT)