A System for Sending the Right Hint at the Right Time

Matthew Elkherj
UCSD
La Jolla, CA, 92037
melkherj@ucsd.edu

Yoav Freund
UCSD
La Jolla, CA, 92037
yfreund@ucsd.edu

ABSTRACT

Hints are sometimes used in online learning system to help students when they are having difficulties. However, in all of the systems we are aware of, the hints are fixed ahead of time and do not depend on the unsuccessful attempts the student has already made. This severely limits the effectiveness of the hints.

We have developed an alternative system for giving hints to students. The main difference is that the system allows an instructor to send a hint to a student *after* the student has made several attempts to solve the problem and failed. After analyzing the student's mistakes, the instructor is better able to understand the problem in the student's thinking and send them a more helpful hint.

We have deployed this system in a probability and statistics course with 176 students. We have demonstrated the superiority of the new hints methodology over the traditional one.

The limiting factor on the effectiveness of our system is the amount of manual labor required to send each hint. This is the main obstacle we see in scaling this approach to larger classes and to MOOCs. We are currently exploring several approaches for addressing this problem: 1) Letting students send hints to their peers. 2) Creating hint libraries. 3) Using machine learning methods to automate the process of mapping student mistakes to the most relevant hint.

Author Keywords

Online Homework; Hints; Real-time intervention

ACM Classification Keywords

K.3.1 COMPUTERS AND EDUCATION: Computer Uses in Education

THE CHALLENGE

Webwork [3] is a popular system for administering homework assignments in mathematics. Webwork has been in use for almost twenty years and is currently used in more than 700 colleges and universities around the world. One of its most attractive aspects is the vast Open Problem Library that contains more than 25,000 problems in various areas of mathematics.

We have been using Webwork in the upper-division undergraduate course: UCSD/CSE103: Introduction to probability and Statistics. The main advantage from the student perspective is receiving immediate feedback on their answers. Many problems are broken into steps, guiding the students how to partition a problem into smaller and easier parts and providing immediate feedback on the answer for each part. Students reaction was positive to the introduction of the webwork system. However, some of the stronger students are actually slowed down by the multi-part questions. They start by finding the final answer and then go back and fill in the steps so that they can get full credit.

On the other end of the spectrum, there is a significant fraction of students that continues to struggle, even with all of the multi-step help. While most students devote to each part less than five minutes and get the correct answer within two or three trials. Struggling students often spend 30 minutes or more on a problem and might eventually give up without finding the correct answer.

Moreover, detailed analysis shows that the struggling students often find an almost-correct answer after a few trials. However, the webwork system is not able to distinguish "almost-correct" from "incorrect" and gives the same feedback to the student. The student seems to lose faith in their own thinking and spends long periods of time in frustration, often never finding the correct answer. (Some of these findings are described in a You-Tube video [1].

ADAPTIVE HINTS

In order to effectively help these struggling students, we developed a system we call "adaptive hints" which is an extension of Webwork [1]. The purpose of the system is to help instructors identify struggling students in real time, identify their closest-to-correct answer, and send them a personalized hint that would help them in a productive direction. The hints that we give are, for the most part, questions. The goal of the question is to help the student become aware of the mistake in their thinking.

Providing hints as part of online problems is a long standing practice. The novelty in our adaptive hints system is that the

[1] we chose to develop our application as an independent that extends webwork, rather than as a tightly integrated extension, so that it would be relatively easy to port the extension to other online education frameworks such as edX.

L@S'14, March 4–5, 2014, Atlanta, Georgia, USA.
ACM 978-1-4503-2669-8/14/03.
http://dx.doi.org/10.1145/2556325.2567864

hints are written *after* student mistakes are observed by an instructor. Standard hints are written together with the problem and represent the single most common mistake that the problem author expects the students to make.

However, our experience shows that student mistakes are hard to predict and often arise from a fundamental conceptual misunderstanding. A good example is the difference between "order matters" and "order does not matter" in combinatorics. Students often believe they understand the concept but their understanding is incorrect. It is only by observing the student's mistakes that such misunderstandings can be revealed and corrected. Our system is therefore closer to a tutoring system, with the advantage of better utilization of tutor time, as one tutor can help 4-5 students at the same time, rather than just one.

RESULTS FROM DEPLOYMENT

We have deployed our system in the "An introduction to probability and statistics" given in the Fall quarter or 2013 in UCSD. This has been the second time that we used Webwork in the course and the first time that we used the adaptive hints. There were 176 students in the class, an instructor (Dr. Freund) three TAs and two tutors. The tutors were both students in the class in Fall 2012, the first time that webwork was used. The adaptive hints system was built and deployed by the TAs.

Feedback from students was very positive. A common reaction was "webwork with hints force you to actually learn".

Here are some statistics that demonstrate the effectiveness of our system. A total of 1897 hints were sent to students. Out of these, the receiving student attempted to answer 792 of the hints and answered correctly 440 of the hints. We refer to the last 440 cases as the "confirmed-impact hints". Out of the 440 confirmed impact hints the final answer provided by the student was correct.

To prove that the hints significantly improved the performance of the student, we wish to reject the null hypothesis stating that the hints have no effect on performance. We use a Monte-Carlo simulation to estimate the statistics under the null hypothesis and compare them to the statistics measured in the experiment. The Monte Carlo estimation randomly picks a sequence of student attempts and the attempt number in which the hint was given (hints are usually given only after the student makes at least 5 unsuccessful attempts.)

Specifically, we test for the following alternative hypotheses:

1. The probability that the student answers a problem part correctly after receiving a confirmed-impact hint is higher than otherwise. Using the 440 confirmed impact cases we get that the probability of getting the final answer correctly is 0.911 while in according to the null hypothesis it is 0.905. This means that the test *failed* and that the effectiveness of the hints cannot be judged by whether or not the students got the final answer correctly.
2. Given that a problem part is eventually answered correctly, the expected number of attempts after a confirmed-imact hint is significantly smaller than otherwise. Using the 401 cases with confirmed hint and a final correct answer and the Welch's t-test for two samples having the same mean, possibly different unknown population variance gives a t-statistic of -4.592 and very significant p-value of 5.527e-06

While our system proved effective for the students that got hints, it proved difficult for a staff of six (instructor, three TAs and two tutors) to produce enough hints to satisfy the need. We estimate that we sent hints to around 5% of the students that could have benefited from them.

FUTURE DIRECTIONS

We know that tutoring through hints is effective. The challenge we now face is how to effectively scale up the system so that all students that need help get effective hints. Our first step in this direction was to create a hint database which allows instructors to reuse, share and improve upon previously written hints. Other directions we plan to pursue are:

- **Automatic assignment of hints:** We plan to use machine learning methods to automate the mapping of incorrect answers to hints. More specifically, we plan to use semi-supervised clustering algorithms to group the mistakes. After the instructor associates a hint with each cluster, we plan to use classification learning algorithm to map mistakes into hints.
- **Empowering students:** In our experience, undergraduate tutors are often the most effective at identifying mistakes and writing hints. Tutors are students that have done well in the class in a previous year. Based on this, we plan to give students that are performing well the possibility of authoring and assigning hints to their fellow students that are struggling. Students that give effective hints (but not by giving the answer) will be rewarded with extra credit points.
- **Integrating With Discussion Boards:** We use the **Piazza** [2] discussion system, and students often discuss individual Webwork problems. Cross linking the problems and the discussion items will help students and instructors effectively distribute questions and answers.
- **The knowledge graph:** Currently hints operate within questions independently, each hint providing help for a specific problem part. As different approach can be to provide the student with a link to an earlier lesson which teaches something they are rusty on. Such links extend the standard linear order of subjects into a "knowledge graph" which represent the pre-requisite relationships between subjects. Linking lessons in this way can help students separate what they need to learn from what they already know.

ACKNOWLEDGEMENTS

The authors are grateful to the National Science Foundation for support under grant IIS-1162581.

REFERENCES

1. Elkherj, M., and Freund, Y. Adaptive hints. You-Tube vidoe, 2013. `http://youtu.be/7KNzBAlh8L0`.
2. Piazza. `https://piazza.com`.
3. Webwork. `http://webwork.maa.org`.