EARLY DETECTION OF AT-RISK STUDENTS IN CS1 USING TEACHBACK/SPINOZA

William T. Tarimo, Fatima Abu Deeb, Timothy J. Hickey
Computer Science Department
Brandeis University
415 South Street – MS 040, Waltham, MA 02453
860-501-7893, 781-392-4222, 617-935-9065
wtarimo, abudeebf, tjhickey@brandeis.edu

ABSTRACT

In this paper, we provide evidence that analysis of student interaction with web-based pedagogical tools in a large partly-flipped CS1 class can be used for early detection of at-risk students. We used student interaction data to estimate five learning style features: engagement, learning speed, confidence, drive, and persistence. We found a positive correlation between final course grades and two of the features of student learning style: engagement and learning speed. Drive and persistence were not good predictors of student success but did provide lower bounds on overall course performance. Confidence was a poor predictor of overall course performance. This kind of analysis provided a more timely and nuanced view of student learning styles than can be obtained from traditional summative assessments and this data could allow for early detection of at-risk students in an introductory programming course.

INTRODUCTION

In this paper, we report on two web-based classroom pedagogical tools (TeachBack and Spinoza) that we used to partially flip a large CS1 class teaching Java programming to novices. TeachBack is an in-classroom web application designed to facilitate various pedagogical interactions for the instructor and students. It is designed to facilitate interactions such as students giving feedback to the instructor, formative assessments, in-class discussions and assistance through back-channel forums. It also provides statistics on student performance in formative assessments and participation. The application is a redesigned newer version of `The Affective Tutor'[6] with added functionality. TeachBack provides three major tools:

- iResponder is a light-weight `clicker' tool for formative assessments which allows the instructor to easily create questions as well as summarize and grade students' responses.
- Forum is a back-channel supervised discussion forum where students can post questions and ideas to the entire class or interact privately with the instructors and TAs. This is an avenue for students to get individualized help without having to ask the instructor and interrupt the class.
- Feedback is a feature that provides a way for students to give affective and cognitive feedback to the instructor throughout an activity or lecture. By default, TeachBack offers three feedback sentiment choices, Engaged, Bored, and Confused plus an optional 50-character comment.

Spinoza [1] is a web-based Java IDE that has been designed to help novice programmers develop competence in Java coding in a flipped classroom. It has two main features:

- SpinozaHomework introduces students to Test Driven Development by providing them a problem, some initial code and a suite of unit tests, some of which are randomly generated each time the program is run.
- SpinozaExercise is a lightweight version of SpinozaHomework that does not provide the automatic unit testing, but can be used to create in-class coding problems on-the-fly to test student understanding of newly introduced concepts.

These tools enable a wide variety of active learning pedagogies and provide a deep, real-time, and nuanced view of student performance and interactions in the class. We discovered, while analyzing the data after the course was complete, that there were several interesting correlations between the way students interacted with these tools and various measures of student performance.

RELATED WORK

Flipped classrooms have been shown to improve learning outcomes by promoting deeper understanding and greater retention of material [2, 11, 5] without having to cover less material [9]. Adopting these new pedagogies, however, requires additional effort on both the teaching and learning sides. Students need to adapt to covering the content on their own before class, and teachers need to embrace the role of being a coach in the classroom, using techniques that engage the students directly in the content. Think/Pair/Share exercises [8] in flipped classrooms provide various forms of active assessment activities by allowing students to work on activities, first individually, then in pairs or small groups and finally as a whole class. This technique allows students to actively assess their learning by trying to solve a problem, reflecting on their learning with a peer and receiving immediate feedback from others in the class. Active learning methods such as Think/Pair/Share can be scaled to larger class sizes when one makes use of computer-mediated communication [10] such as clickers, backchannel forums and other tools that support student-student and student-teacher online communication.

TeachBack and Spinoza were developed to facilitate interactive classroom activities such as individual and cooperative formative assessments [3]. In this paper, we show that they provide an additional unforeseen advantage - the analysis of student interactions with these tools provides a deep insight into student learning styles and this knowledge can help us predict which students may do poorly in the class.

METHODOLOGY

During the fall semester in 2014, we taught an Introduction to Programming in Java class which had a total of 284 students, divided into two roughly equal sections, each meeting three times a week for 50 minutes. In this class, we employed an active learning style of pedagogy. The two sections of the class were taught using the same interactive pedagogy where each class meeting started with pre-class assigned readings and a mandatory reading reflection or short quiz due before class. The 50-minute class sessions were composed of 3-5 interactive activities including short lectures, written feedback from students, clicker-based formative assessments, and coding challenges, often using a Think/Pair/Share methodology [8]. TeachBack and Spinoza were used in almost every class to facilitate these activities and communications between the instructor and the students.

All student interactions with TeachBack and Spinoza were recorded in a database. This data was then combined with traditional grade book information including scores on homeworks, quizzes and responses to four surveys administered throughout the semester. For this study, we looked for features of student interaction that were correlated with overall performance in the course. Rather than use the letter grade, we measure overall course performance with an uncurved final course score that combines quiz scores, homework scores and participation. A score under 60% indicates a poor performance in the course and our goal was to seek student interaction data from the TeachBack and Spinoza tools that would help us identify students in danger of doing poorly in the class as early as possible so that some intervention could be taken.

RESULTS AND DISCUSSION

Engagement and Performance

The simplest measure of engagement in a course is whether students are attending class and participating. We calculated each student's total engagement by adding three quantities: (1) the number of questions attempted in iResponder, whether they were answered correctly or incorrectly, (2) the total number of feedback responses, and (3) the total number of forum posts and comments. Our first result is that total participation in TeachBack is positively correlated with final course score. The data is plotted in Figure 1 together with a trend-line. This measure could be helpful in predicting which students might be at-risk, since we can observe that all students who performed poorly in the class (with a course score under 60%) had a participation level under 60. Students were required to answer at least half of all iResponder questions and their participation accounted for 10% of their final course grade so we would expect at least a little correlation between participation and course grade.

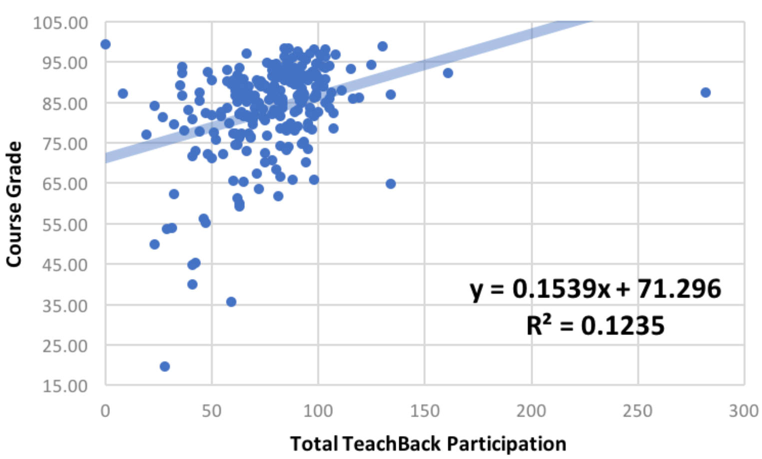

Figure 1: Engagement: Course grade vs. total TeachBack participation.

Learning Speed and Performance

The next measure we studied was how quickly students could assimilate new knowledge, which was measured as the sum of all iResponder points. The iResponder data provides a formative assessment which is a rough measure of the initial level of student understanding. Since iResponder questions are graded (typically on a 0-2 scale), the sum of all of their iResponder points over the semester gives a measure that combines participation and formative assessment of initial levels of understanding.

Since iResponder questions were often asked as soon as a new concept was introduced, one would not expect a perfect correspondence between this formative measure and the summative measures used to calculate the overall course grade. We found however, that total iResponder points, as a proxy for learning speed, is strongly correlated with overall performance in the course. Figure 2 shows a plot of total iResponder points versus uncurved final course score. This plot shows that performance in formative assessments is even more strongly correlated to the final course score than engagement and this indicates that fast learners tend to do better in the course and all students who did poorly had under 40 iResponder points. It could be that students who score high on total iResponder points are those students who are being challenged in their Zone of Proximal Development [12]. These findings demonstrate that student performance and initial understanding in class can roughly predict their overall learning and grades.

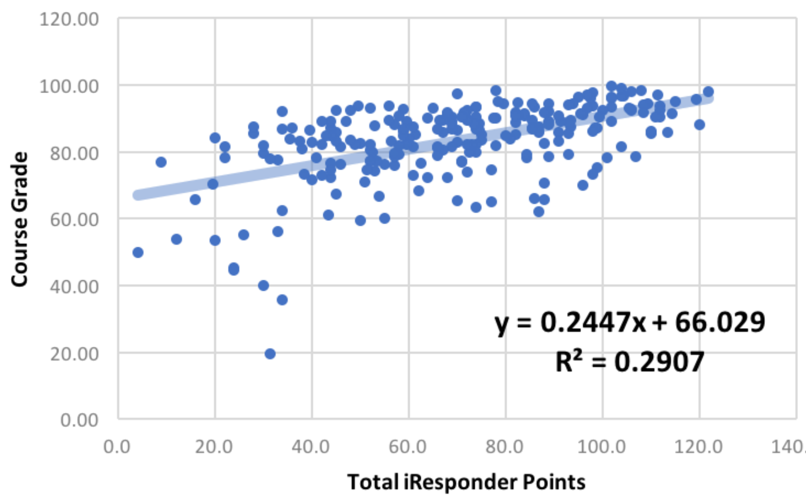

Figure 2: Learning Speed: Course grade vs. total iResponder points.

Confidence and Performance

A third measure that one might think could predict performance is student confidence which we measure as the student self-assessed level of confusion when first encountering new topics. One might think that students who feel they are often confused in class would be in danger of doing poorly. To get a measure of self-assessed initial confusion, we examined all student feedback responses and summed them with each confused response counted as -1 and each bored or engaged response counted as 1. Figure 3 contains a scatter plot of confidence versus course grade and we can see that the R^2 level is very low at 0.02 which indicates that the correlation is very weak. Indeed, we can see that the students with very low confidence levels are not the students who did poorly in the class. At-risk students seem to be overconfident in their level of understanding [7]. The students who were most confused, generally did fairly well in the course and we suspect this might be due to imposter syndrome where students who do not fit the cultural stereotype underestimate their ability [4].

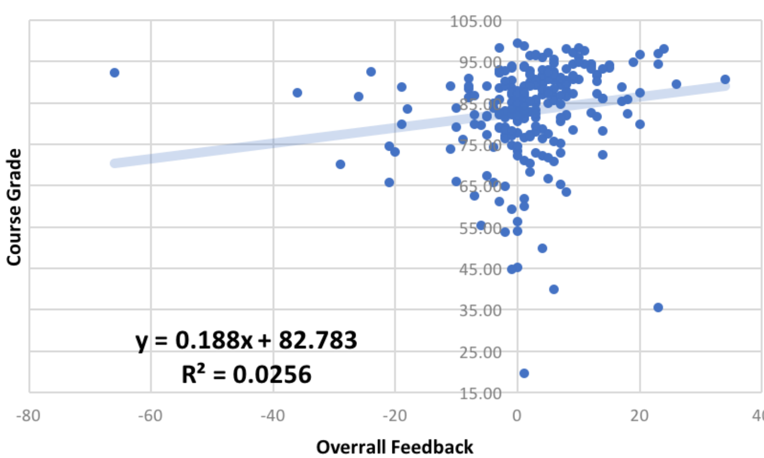

Figure 3: Confidense: Course grade vs. self-assessment based on feedback.

Drive and Performance

One way to measure student drive in this CS1 class is to count the number of SpinozaExercise problems they attempt. This includes both in-class coding activities as well as optional activities completed at home in preparation for each of the four quizzes. We found a very slight positive correlation between the total number of problems a student attempted and their course grade. This is shown in Figure 4.

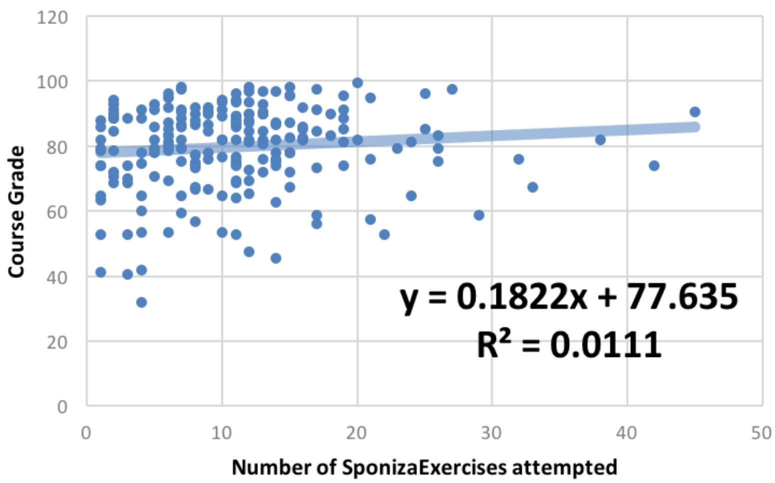

Figure 4: Drive: Course grade vs. number of exercises attempted.

Persistence and Performance

Another feature of student commitment to the course is their persistence, which can be estimated by measuring the amount of time they continue to invest in challenging problems. One advantage of requiring students to use Spinoza for their coding is that we have a record of each version of the code they developed rather than just the final product which is typically all that an instructor receives in a traditional CS1 course. To measure persistence, we look at the average number of attempts that a student makes on SpinozaHomework problems by measuring the average number of times they press the Run button to compile and run their problem code. Figure 5 shows the relationship between course grade and persistence (i.e. average number of attempts for homeworks completed with the SpinozaHomework tool).

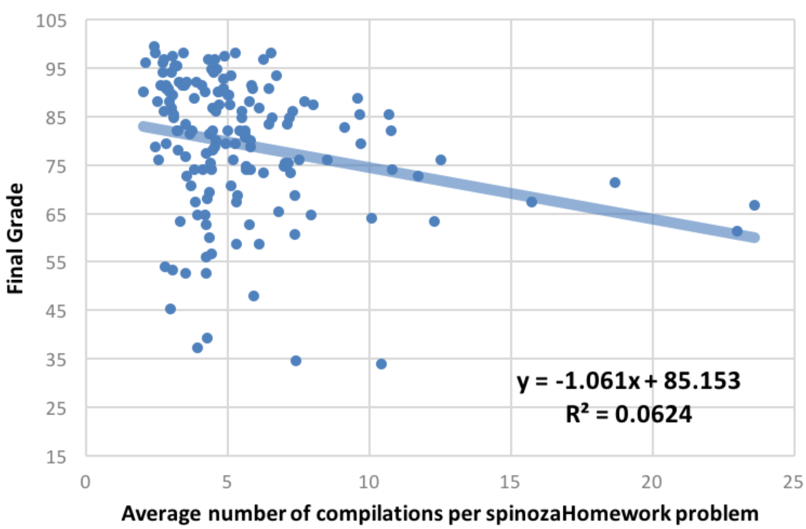

Figure 5: Persistence: Course grade vs. number of compilations per homework problem.

Most students make between 2 and 7 attempts on each problem. The students who do well in the course (course grade > 90%) are able to solve the problems with a reasonable effort. Likewise, the students who do poorly (course grade < 60%) also expend at most a moderate effort, but then usually give up without completing the problems. The students in the mid-range (60-90%) tend to put in more effort per problem and a few near the bottom of the acceptable range expend a great deal of effort with an average of up to 30 attempts on each problem, indicating a high level of persistence. The students who have the highest number of attempts per problem tend to be in the high C range (near 70), working hard, not understanding fully, but not failing. The at-risk students tend to have a very low number of attempts per problem, perhaps because they give up out of frustration before completing the problem. One interesting observation here is that that persistence, as measured here, provides a lower bound for homework and course grade which suggest that if we can increase persistence in at-risk students that they can do well on homework problems.

CONCLUSION AND FUTURE WORK

This study has shown that there are several strong correlations between the way students interact with the tools TeachBack and Spinoza and their overall performance in the course. Moreover, selected measurements provide proxies for student features that are difficult to measure through standard summative assessment means. We suspect that by monitoring student interactions with these tools, one might be able to detect the students who are in danger of doing poorly as soon as they start to have trouble in the course, and well before the standard summative measures such as quiz and homework scores indicate a problem. In a class where TeachBack and Spinoza are used everyday this would provide a sensitive, early-warning system to help direct TA resources towards students that would benefit from the extra support. We intend to add these detection features to the next versions of TeachBack and Spinoza and test these hypotheses in a future class.

REFERENCES

[1] Abu Deeb, F., Hickey, T. J., Spinoza: The Code Tutor. *Proceedings of the 2015 International Conference on Computer and Information Science and Technology.*

[2] Amresh, A., Carberry, A. R., Femiani, J., Evaluating the effectiveness of flipped classrooms for teaching CS1. In *Frontiers in Education Conference, 2013 IEEE* (pp. 733-735). IEEE.

[3] Angelo, T. A., Cross, K. P., Classroom assessment techniques. (1993).

[4] Cozzarelli, C., Major, B., Exploring the validity of the impostor phenomenon. *Journal of Social and Clinical Psychology* 9.4 (1990): 401-417.

[5] Hauswirth, M., Adamoli, A., "Teaching Java programming with the Informa clicker system." *Science of Computer Programming* 78.5 (2013): 499-520.

[6] Hickey, T. J., Tarimo, W. T., The affective tutor. *Journal of Computing Sciences in Colleges* 29.6 (2014): 50-56.

[7] Koriat, A., Bjork, R. A., Illusions of competence in monitoring one's knowledge during study. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 31.2 (2005): 187.

[8] Kothiyal, A., Murthy, S., Iyer, S., Think-pair-share in a large CS1 class: does learning really happen?. *Proceedings of the 2014 conference on Innovation & technology in computer science education*. ACM, 2014.

[9] Nelson, C. E., On the persistence of unicorns: The trade-off between content and critical thinking revisited. *The social worlds of higher education: Handbook for teaching in a new century* (1999): 168-184.

[10] Reed, A., Computer-mediated communication (CMC) and the traditional classroom. *Teaching with Technology Today* 5.6 (2000).

[11] Sarawagi, N., Flipping an introductory programming course: yes you can!. *Journal of Computing Sciences in Colleges* 28.6 (2013): 186-188.

[12] Vygotsky, L., Interaction between learning and development. *Readings on the development of children* 23.3 (1978): 34-41.