Clustering and Comparing Students' Interactions with an eTextbook

Mohammed Farghally (Virginia Tech, mfseddik@vt.edu)
Mostafa Mohammed (University at Buffalo, mkomohmd@buffalo.edu)

1 Background

Online interactive eTextbooks have been used extensively to teach students core Computer Science (CS) topics. As a result of students' interactions with eTextbooks, massive amounts of data are generated and logged awaiting further meaningful analysis. The OpenDSA project [4] provides the required infrastructure and content necessary to create a collection of online, open-source tutorials combining textbook quality text with interactive visualizations and exercises with automated feedback. OpenDSA supports the collection and recording of students' clickstream data resulted from their interaction with the prose, visualizations, and exercises [1, 4]. In a typical course offering with a few hundreds of enrollments, millions of records are generated and stored in OpenDSA's backend. Better instructional experiences could be achieved by making use of this data and understanding students' learning behaviors by analyzing their interactions with different eTextbook learning content.

2 Preliminary Work

OpenDSA students' interactions are stored in a raw event-level form. Accordingly, it is difficult for developers or instructors to understand student behaviors, or to verify hypotheses about relationships between behavior and course performance. Our prior work focused on abstracting raw event-level interactions (e.g., page load, button click, page unload) into study sessions (a series of continuous interactions by a student without a significant time gap) [7]. We collected clickstream data from the OpenDSA platform for a data structures and algorithm course offered during Spring 2024 at Virginia Tech, and we segmented the continuous clickstream into study sessions. After that, we abstracted sessions into higher-level semantic states like reading a prose, interacting with a visualization, or answering an exercise, along with the number of transitions between each state. This prior work was a useful step towards facilitating the creation of a general analysis pipeline.

3 Objective

Our goal in this project is to categorize students' interactions with an eTextbook created using the OpenDSA infrastructure for a data structures and algorithms course offered during Spring 2024 at Virginia Tech. Each category will represent a distinct study profile. This categorization will be based on features extracted from students' interaction logs like the time spent interacting with visualizations, the time spent solving exercises, the time spent reading textual content, and the frequency of transitions between prose, visualizations, and exercises. After that, we will verify whether there is any effect of the type of study behavior on students' final grade in the course, and which category(s) contribute to higher grades. By achieving this goal, eTextbook developers can then redesign their content to require students to follow the study behaviors that contribute to higher grades.

In this project, we build over our prelinminary work and start clustering those abstracted study sessions to answer the following research questions:

1. Based on features like time spent reading a prose, time spent interacting with a visualization, time spent solving exercises, and the frequency of transitions between different artifacts of study content, what are the different categories of students' study behaviors as they interact with different eTextbook content?

2. Is there any effect of the study behaviors on students' performance as measured using final course grades?

Prior research have shown that visualizations could be an effective way of teaching abstract CS concepts as it better engages students into the content when compared to traditional prose material [2]. Earlier research reported that students will get better improvement in their learning when they spend enough time answering questions in addition to interacting with visualizations [5]. For example, in [8], students who answered questions integrated into a set of visualizations received better scores on a final exam in a formal languages course when compared to students who just used the visualizations. In [6], students who spent more time solving recursion programming exercises received significantly better scores on a final exam.

Based on this, there might be an assumption that those students who spend more time solving exercises or those who solve more questions will receive better course grades. However, in [3] it was found that the majority of OpenDSA students are credit seekers skipping directly to the exercises then go back to the visualizations or prose content if they don't know the answer. In our analysis, we decided to also include the frequency of transitions between different learning content (prose, visualizations, and exercises), and see if this will have an effect on students' scores in addition to the time spent in each learning content.

4 Deliverables

We aim to have the following deliverables out of this project:

- A detailed technical document showing how raw OpenDSA logs are abstracted into higher level study sessions.
- A tool to extract anonymized students' interaction logs for a specific eTextbook from the OpenDSA database.
- Have the extracted data transferred to and publicly available on DataShop [1].
- An analysis tool within Datashop to process the logs and produce the different study behavior categories.
- A publishable research paper describing the technical and research aspects of the work.

5 Alignment with SPLICE goals

We believe that our work aligns well with SPLICE goals for two reasons. First, our work is based on a well-known existing infrastructure that supports the collection of students' learning data, and has been used before in Computing Education Research (CER) [2, 6, 8]. Second, our goal in this project is to develop an analysis mechanism to make use of the collected learning data in answering research questions and verifying hypotheses. We believe that our results from this work will be helpful to the whole CER community including tools and infrastructure developers, instructors, and researchers. All of the deliverables mentioned in Section 4 will be publicly available, so that others could make use of it. In addition, we plan to publish a research paper out of this work.

[1] https://pslcdatashop.web.cmu.edu/

6 Project team

- Dr. Mohammed Farghally is a Collegiate Assistant Professor of Computer Science at Virginia Tech. Dr. Farghally's research interests include digital education, learning analytics, and CS ethics. Dr. Farghally has an extensive teaching experience, and remarkable contributions to the CER community.
- Dr. Mostafa Mohammed is a CS Assistant Professor of Teaching at the University at Buffalo. Dr. Mohammed received the instructor of record teaching excellence award from Virginia Tech. Dr. Mohammed has significant contributions to the CER community. Dr. Mohammed relies extensively on OpenDSA in his teaching, so he would be a valuable resource in guiding the project.

7 Budget

We request a total of $10,000 to support this project. $8000 will be used to hire a graduate research assistant to work on the project for one semester, and the remaining $2000 will be used to support conference travel to present our work.

References

[1] Daniel Aubrey Breakiron. Evaluating the integration of online, interactive tutorials into a data structures and algorithms course. Master's thesis, Virginia Tech, 2013.

[2] Mohammed F Farghally, Kyu Han Koh, Hossameldin Shahin, and Clifford A Shaffer. Evaluating the effectiveness of algorithm analysis visualizations. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pages 201–206, 2017.

[3] Eric Fouh, Daniel A Breakiron, Sally Hamouda, Mohammed F Farghally, and Clifford A Shaffer. Exploring students learning behavior with an interactive etextbook in computer science courses. *Computers in Human Behavior*, 41:478–485, 2014.

[4] Eric Fouh, Ville Karavirta, Daniel A Breakiron, Sally Hamouda, Simin Hall, Thomas L Naps, and Clifford A Shaffer. Design and architecture of an interactive etextbook–The OpenDSA system. *Science of Computer Programming*, 88:22–40, 2014.

[5] Scott Grissom, Myles F McNally, and Tom Naps. Algorithm visualization in cs education: comparing levels of student engagement. In *Proceedings of the 2003 ACM symposium on Software visualization*, pages 87–94, 2003.

[6] Sally Hamouda, Stephen H Edwards, Hicham G Elmongui, Jeremy V Ernst, and Clifford A Shaffer. Recurtutor: An interactive tutorial for learning recursion. *ACM Transactions on Computing Education (TOCE)*, 19(1):1–25, 2018.

[7] Samnyeong Heo. Analyzing student session data in an etextbook. Master's thesis, Virginia Tech, 2022.

[8] Mostafa Mohammed, Clifford A Shaffer, and Susan H Rodger. Teaching formal languages with visualizations and auto-graded exercises. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, pages 569–575, 2021.