Integrated "Caring" IDE: A CS1 Tool

Mohsen Dorodchi
UNC Charlotte
Charlotte, NC, USA
Mohsen.Dorodchi@uncc.edu

Aileen Benedict
UNC Charlotte
Charlotte, NC, USA
abenedi3@uncc.edu

Erfan Al-Hossami
UNC Charlotte
Charlotte, NC, USA
ealhossa@uncc.edu

ABSTRACT

In this work, we explain the design and implementation of a collaborative programming environment capable of (1) collecting various forms of data from students both directly and indirectly, (2) organizing and visualizing that data, and most importantly (3) providing statistical and predictive analytics features. This system integrates with learning management systems (LMS) through various forms of API's and therefore, helps instructors track students' progress in introductory programming courses and predict any potential failures early in the semester. The integration of emergent social tools for real-time communication and assessment, such as live chat, blogs, discussion boards, personal response systems, and surveys, would bring new and innovative capabilities to the classroom, especially for active learning models. A cloud-based integrated development environment (IDE) is being integrated and is under final tests that track students' programming patterns both in individual activities as well as group, collaborative ones. The seamless integration with learning management systems can collect a great amount of data about students' study patterns beyond grades as well. All of this data is descriptively visualized in different granular levels. In addition, they are fed into the learning analytics module to determine at-risk students early on. In this paper, we will report on the layout of the system as well as various tools that have been experimented with in our CS1 active learning course fostering students' engagement and learning. Most parts of the system are currently built and functional. Ultimately, the goal is to expand this infrastructure to different institutions through collaborative research.

CCS CONCEPTS

CS1; Human-Centered Computing~Visualization Toolkits

KEYWORDS

CS1; IDE; Learning Analytics

INTRODUCTION

There has been an increased interest in learning analytics due to the availability and increased ease of collecting data of the students' learning process [4]. Learning analytics provides a process to collect, measure, analyze, and reflect on data. The goal of which is to understand and optimize student learning and the environments in which it occurs. When such data is available, analytics, and visualizations methods can be used to help the educators as well as students to gain valuable insights into students' learning. These insights can then be used to make decisions on how to modify courses to help students, such as designing student interventions during the course, adapting coursework (such as activities, lectures, or group discussions) to the students' learning needs, and evolving the overall course design over the course of semesters of offerings.

In this paper we discuss our system which (1) collects data from various sources, such as students' performance and participation through the LMS API, students' reflections, and live chat, (2) analyzes the data, including statistical and predictive modeling, and (3) visualizes the data and results.

RESEARCH DESIGN

In this section we describe the part of the system which oversees data collection, analytics, and visualization. Figure 1 showcases the architecture of the analytics module of the system which includes two major submodules as indicated by dashed lines: A) the data collection and preparation and B) the analytics and visualizations.

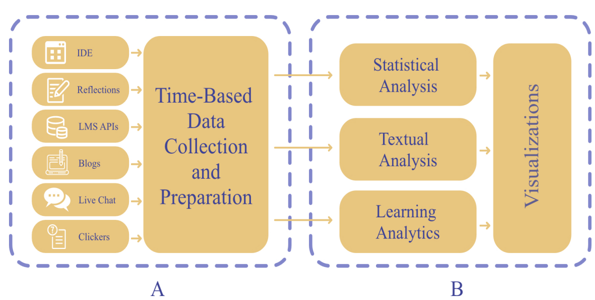

Figure 1. The analytics module.

A) Data Collection and Preparation

This submodule collects detailed students' data throughout the semester from multiple different sources as indicated.

Each data entry is encoded with a timestamp and stored accordingly. The IDE data includes compilation information (syntax errors, etc.) as well as execution data such as number of runs, number of errors and types of errors. Students' reflections are collected through surveys at the beginning and/or during, and/or end of activities. Students' blogs are coming from a separate blog site, and students normally complete the blogs biweekly based on guided prompts about their learning and teamwork. Blog posts are scraped and encoded with a timestamp and then stored. Students were also instructed to write a document for extra credit on topics or concepts that were challenging for them in the class (muddiest point). Student data such as grades, participation and activity are retrieved from the LMS using API calls. Students chats transcripts are automatically processed and fed into the system. Lastly, PRS results are exported into the LMS system as well as our system.

B) Analytics and Visualizations

Our proposed system provides the following analytics: 1) descriptive statistics, 2) textual analytics, and 3) learning analytics. Statistics provide a basic tales of mean, median, standard deviation and mode. As far as approaches, we used bag-of-words, log-odds ratio [3], LIWC [5] and topic modeling.

RESULTS AND DISCUSSION

In this section, we present sample outputs of our proposed system. Figure 2 shows a log-odds ratio visualization of our course reflections, an index signifying the likelihood of a word belonging to a certain group over another one [3]. In our case, the groups were separated by gender (Male/Female). The x-axis represents the word frequency in the dataset. and the y-axis represents the log-odds ratio. The word 'activity' is more likely to be used by females over males with a log-odds ratio of 3.6.

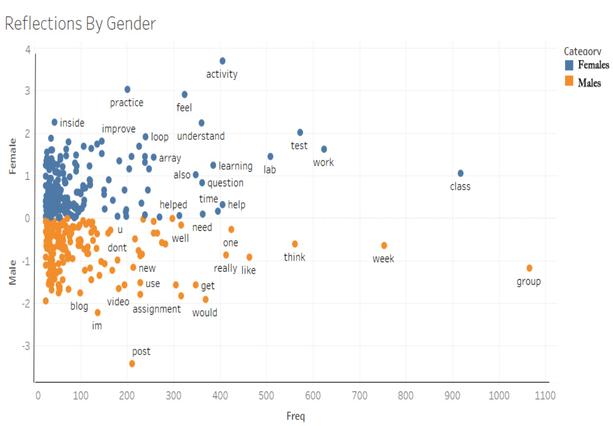

Figure 2. Visualization representing word belonging likelihoods

NMF		LDA	
Topic 0:	helped figure problems	**Topic 0:**	able partner help
Topic 1:	partner work able	**Topic 1:**	worked mistakes didn't
Topic 2:	yes efficient did	**Topic 2:**	did things yes
Topic 3:	ideas bounce sharpen	**Topic 3:**	programming syntax correct
Topic 4:	did help peer	**Topic 4:**	helped understand didn't
Topic 5:	helpful opinions having	**Topic 5:**	ideas great bounce
Topic 6:	Didn't peer know	**Topic 6:**	helpful ways got

Table 1. Topics of lab 8 peer programming Questionnaire

After completing the last major lab activity, where students use multidimensional arrays significantly, around 296 students were completed a reflection following the lab. Table 1 shows the outputs of two different topic modeling algorithms NMF and LDA on the question of "How did Peer programming help you in today's session?" It can be interpreted from such analysis that peers bounce different ideas off while helping each other and collaborate on fixing mistakes.

REFERENCES

[1] M. Dorodchi, A. Benedict, D. Desai, M. J. Mahzoon, S. MacNeil, N. Dehbozorgi. (2018). Design and Implementation of an Activity-Based Introductory Computer Science Course (CS1) with Periodic Reflections Validated by Learning Analytics. FIE 2018.

[2] M. Dorodchi, A. Benedict, and E. Al-Hossami. (2019). CS1 Scaffolded Activities: The Rise of Students' Engagement. In Proceedings of the 2019 ACM Conference on International Computing Education Research (ICER '19). ACM, New York, NY, USA.

[3] B. L. Monroe, M. P. Colaresi, and K. M. Quinn. (2008). Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. Political Analysis 16.4 (2008): 372-403.

[4] C. D. Hundhausen, D. M. Olivares, and A. S. Carter. (2017). IDE-based learning analytics for computing education: a process model, critical review, and research agenda. ACM Transactions on Computing Education (TOCE) 17.3 (2017): 11.

[5] J. W. Pennebaker, M. E. Francis, and R. J. Booth. (2001). Linguistic inquiry and word count: LIWC 2001. Mahway: Lawrence Erlbaum Associates 71.2001 (2001): 2001.