Augmenting Authentic Data Science Environments for Learning Analytics

Anant Mittal
School of Information, University of Michigan
anmittal@umich.edu

Christopher Brooks
School of Information, University of Michigan
brooksch@umich.edu

ABSTRACT: Unlike general learning management systems which have used fine-grained trace behaviours to understand learning processes, data science environments use discipline-specific tools such as Project Jupyter (Perez and Granger, 2015). Augmentation of these tools is necessary in order to surface learner activities in ways which might be used for adaptation (Ferguson, 2012). This is analogous to augmenting problem-solving environments for mathematics (Melis and Siekmann, 2004), where domain-specific tools are necessary for understanding learning activity. In this work, we specifically tackle the augmentation of Project Jupyter. We explain the architecture of the environment along with the types of events we are able to collect and frame research questions we aim to answer with this work.

Keywords: Data science education, Jupyter Notebook, Data Mining, Learning Analytics

1 OVERVIEW OF JUPYTER LOGGING

Project Jupyter, the *de facto* standard learning environment for python data science education, allows developers to extend its functionality through extensions (Perez and Granger, 2015; Kluyver et al., 2016). In order to capture fine-grained activity of learners, we have created an event-based schema of meaningful actions within the notebooks, and then created JavaScript-based extensions which record student activities such as cell insertions, cell executions, and cell deletions[1]. These extensions log data in JavaScript Object Notation (JSON) and send them to a webservice back-end. The backend APIs use AWS Kinesis Data Streams with Lambda and S3 to provide a serverless endpoint for learning analytics data collection, allowing us to scale to large numbers of learners with minimal infrastructure costs. The extensions are being deployed in Massive Open Online Courses (MOOCs) which use Jupyter as the source for both course assignments and lecture materials.

We capture five kinds of events: when a notebook has been opened, when a notebook has been scrolled within, when a notebook has been saved, when a notebook cell has been executed, and when cell execution has been finished. For each event we capture high level common data, including the course context where the notebook is deployed (e.g. the assignment or weekly lecture context),

[1] A cell in data science education is similar to a stanza in a poem or paragraph in an essay, and tends to encapsulate a single idea or investigation of the student.

an identifier for the student, a timestamp, and metadata of the notebook. We add to this event-specific metadata, such as which cell in a notebook is being manipulated.

2 LEARNING ANALYTICS IN DATA SCIENCE EDUCATION

We aim to mine the granular student activity data we collect, and we intend to focus on tackling multiple overarching concerns in MOOC environments such as student evaluation and a lack of immediate feedback (Hew & Cheung, 2014). Our environment can help instructors and researchers in understanding student's learning behavior and learning outcomes and help them with more active feedback. Specific investigations this infrastructure will help us understand include:

- What are the common student misconceptions in assignments? For instance, with execution cell events we can identify if a student is struggling on a specific question and provide individual feedback, thus reducing student frustration while scaffolding learning with individual help.
- Are students following along with instructional video? Notebooks for all of the videos are available to students, but at the moment it is unclear how they use these notebooks along with video lectures. Through analysis of cell execution timestamps and the clickstream information (e.g. video heartbeat functions), we should be able to determine if students are following along and practicing as they observe the lectures.
- Do students feel more engaged when given immediate feedback? Through program analysis techniques (e.g. source code analysis), we can identify places where we might provide feedback to the students after their cells have been executed, allowing for just-in-time interventions of learning.

As the online education space continues to grow rapidly, institutions need to see learning analytics and educational data mining as a tool to achieve better learning results. For courses (traditional and online) which use Jupyter for assignments, our extensions to the tool can help instructors proactively monitor student performance, identify students at the risk of dropping out, and implement strategies to improve student engagement.

REFERENCES

Ferguson, R. (2012). Learning analytics: drivers, developments and challenges. *International Journal of Technology Enhanced Learning*, *4*(5/6), 304-317.

Hew, K. F., & Cheung, W. S. (2014). Students' and instructors' use of massive open online courses (MOOCs): Motivations and challenges. *Educational research review*, *12*, 45-58.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., ... & Ivanov, P. (2016, May). Jupyter Notebooks-a publishing format for reproducible computational workflows. In ELPUB (pp. 87-90).

Melis, E., & Siekmann, J. (2004, June). Activemath: An intelligent tutoring system for mathematics. In International Conference on Artificial Intelligence and Soft Computing (pp. 91-101). Springer, Berlin, Heidelberg.

Perez, F., & Granger, B. E. (2015). Project Jupyter: Computational narratives as the engine of collaborative data science. Retrieved September, 11, 207.