

SPLICE Mini-Grant Proposal

LearnSphere Workflow Components

Paulo Carvalho & Michael Asher
(Cognition, Learning and Tech Research Group;
Carnegie Mellon University, Pittsburgh)

Arun-Balajiee Lekshmi-Narayanan
(PhD Student, University of Pittsburgh)

The purpose of this work is to build LearnSphere components as well as an example LearnSphere Workflow to extract measures of student resource use in online courses and its impact on learning outcomes. The Workflow will be incorporated as a chain of Components that will consider specific types of files from DataShop – such as student interaction logs in an online digital textbook platform, online MOOC system – as sources to convert into processed files by extracting required variables and measures. In this way, we present a system that can simplify and standardize common types of preprocessing techniques performed to set up the analyses presented in well-cited papers from the literature, into one single pipeline system. The Workflow takes a raw data file made available through DataShop and executes the program logic through sequential components in the environment and is available for users to work without saving files locally. The system will also present various snapshots of data files from the different stages of the pipeline for the users to view or debug. This provides a certain level of modularity for them to consider using a source data file pre-processed up to any specific snapshot along the pipeline.

This work extends existing components and workflows that perform similar analyses but work only with data from OLI to work with more general data stored in DataShop. Our goal is to consider a variety of datasets made available on DataShop, such as the RuneStone dataset (datasetId=4689). In these datasets, the definitions for pages are different and the user will be provided with options to indicate what they consider to be the markers for a page, activity, and other events in the log data for special student interactions in the learning environment. The goal of this work is to extend the robustness of the implementation of the doer effect analyses and components in LearnSphere to more datasets on DataShop.

As a use case for this problem and an example Workflow, we consider a dataset for an online course in programming for introductory programming in JAVA in computer science in the RuneStone intelligent textbooks. The dataset is available on DataShop (datasetId=4689). We aim to understand two research questions related to the doer effect:

- 1) Is solving more problems more strongly related to learning outcomes than reading more pages or watching more videos?
- 2) Does the doer effect interact with students' career and goal-orientation?

We consider the following list of variables which will be processed from the raw datashop file. The following is a list of variables that will be generated at the end of the final stage of the pipeline. In our use case, we will present that these variables can be directly supplied to analytical tools like linear regression to estimate real-world phenomena such as the doer effect and student persistence on MOOCs.

Variable	Value	Description
Anon.Student.Id	Student id	Primary grouping variable
Page Id / page	Page number in the book as counted by page counter	Increment page count that has content (video or activity)
page_w_reading	If page has content other than video or activity	if video + activity time is more than 95% of time on page, then remove that page from page-count
video counter	counts video by the participant at a given page	Increment the video counter that has "video" selection / action corresponding to a page.
Act counter	Counts video by the participant at a given page	Increment the act counter that has "activity" of any of the following types – active_code, live_code, parsons, mcqs, codepens, "clickable area"
video_engagement_agg_t_s	Agg by page for a given student for number of seconds played on a video	Group by page counter then group by student and sum the differences of (v_stop - vstart)
debugging_practice_t_s	Time Spent on codepens,	Group by student group by page select actions that are called "codepens" or "activecode" or "livecode" and agg by time in s
System_feedback	"Unittest" – number of test cases passed Ac_error – syntax error in programming activities	For unittest provide agg group by student by page to total number of successes for ac_error provide agg group by student by page to total number of errors
writing_code_t_s	Activecode, Livecode – writing code in field	Group by student group by page select actions that are called "livecode" or "activecode" and agg by time in s

programming_puzzles_t_s	Time Spent on parsonsMove, parsons, clickableArea,	Group by student group by page select actions that are called "parsonsMove" or "parsons" or "clickableArea" or "dragNdrop" and agg by time in s
programming_quiz_t_s	Time Spent on "timedExam", – exam with 20 MCQs "mChoice", "Shortanswer", "Fillb", "Poll", dragNdrop	Group by student group by page select actions that are called "timedExam" or "mChoice" or "Shortanswer" or "Fillb" or "Poll" or dragNdrop and agg by time in s