MOOCdb: Developing Data Standards for MOOC Data Science

Kalyan Veeramachaneni, Franck Dernoncourt,
Colin Taylor, Zachary Pardos, and Una-May O'Reilly

Massachusetts Institute of Technology, USA.
{kalyan,francky,colin_t,zp,unamay}@csail.mit.edu

1 Introduction

Our team has been conducting research related to mining information, building models, and interpreting data from the inaugural course offered by edX, *6.002x: Circuits and Electronics*, since the Fall of 2012. This involves a set of steps, undertaken in most data science studies, which entails positing a hypothesis, assembling data and features (aka properties, covariates, explanatory variables, decision variables), identifying response variables, building a statistical model then validating, inspecting and interpreting the model. In our domain, and others like it that require behavioral analyses of an online setting, a great majority of the effort (in our case approximately 70%) is spent assembling the data and formulating the features, while, rather ironically, the model building exercise takes relatively less time. As we advance to analyzing cross-course data, it has become apparent that our algorithms which deal with data assembly and feature engineering lack cross-course generality. This is not a fault of our software design. The lack of generality reflects the diverse, ad hoc data schemas we have adopted for each course. These schemas partially result because some of the courses are being offered for the first time and it is the first time behavioral data has been collected. As well, they arise from initial investigations taking a local perspective on each course rather than a global one extending across multiple courses.

In this position paper, we advocate harmonizing and unifying disparate "raw" data formats by establishing an open-ended standard data description to be adopted by the entire education science MOOC oriented community. The concept requires a schema and an encompassing standard which avoid any assumption of data sharing. It needs to support a means of sharing *how the data is extracted, conditioned and analyzed.*

Sharing scripts which prepare data for models, rather than data itself, will not only help mitigate privacy concerns but it will also provide a means of facilitating intra and inter-platform collaboration. For example, two researchers, one with data from a MOOC course on one platform and another with data from another platform, should be able to decide upon a set of variables, share scripts that can extract them, each independently derive results on their own data, and then compare and iterate to reach conclusions that are cross-platform as well as cross-course. In a practical sense, our goal is a standard facilitating insights

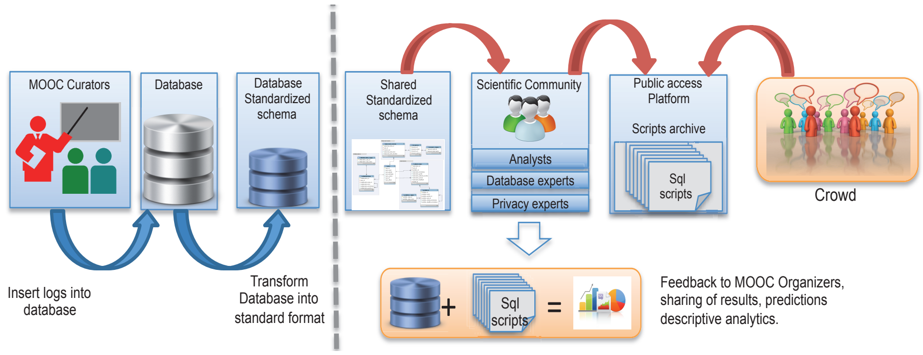
MOOC Curators
Database
Database Standardized schema
Insert logs into database
Transform Database into standard format
Shared Standardized schema
Scientific Community
Analysts
Database experts
Privacy experts
Public access Platform
Scripts archive
Sql scripts
Crowd
+
Sql scripts
=
Feedback to MOOC Organizers, sharing of results, predictions descriptive analytics.

Fig. 1. This flowchart represents the context of a standardized database schema. From left to right: Curators of MOOC course format the raw transaction logs into the schema and populate either private or public databases. This raw database is transformed into a standard schema accepted by the community, (like the one proposed in this paper) and is exposed to the analytics community, mostly researchers, who develop and share scripts, based upon it. The scripts are capable of extracting study data from any schema-based database, visualizing it, conditioning it into model variables and/or otherwise examining it. The schema is unifying while the scripts are the vehicle for cross-institution research collaboration and direct experimental comparison.

from data being shared without data being exchanged. It will also enable research authors to release a method for recreating the variables they report using in their published experiments.

Our contention is that the MOOC data mining community - from all branches of educational research, should act immediately to engage in consensus driven discussions toward a means of standardizing data schema and building technology enablers for collaborating on data science via sharing scripts, results in a practical, directly comparable and reproducible way. It is important to take initial steps now. We have the timely opportunity to avoid the data integration chaos that has arisen in fields like health care where large legacy data, complex government regulations and personal privacy concerns are starting to thwart scientific progress and stymy access to data. In this contribution, we propose a standardized, cross-course, cross-platform, database schema which we name as *"MOOCdb"*. [1]

We proceed by describing our concept and what it offers in more detail in Section 2. Section 3 details our proposed the data schema systematically. Section 4 shows, with a use case, how the schema is expressive, supportive and reusable. Section 5 concludes and introduces our current work.

[1] We would like to use the MOOCshop as a venue for introducing it and offering it up for discussion and feedback. We also hope to enlist like minded researchers willing to work on moving the concept forward in an organized fashion, with plenty of community engagement.

2 Our Concept and What it Offers

Our concept is described as follows, and as per Figure 1:

- It identifies two kinds of primary actors in the MOOC eco-system: *curators* and *analysts*. Curators collect raw behavioral data expressing MOOC students' interaction with online course material and then transfer it to a database, often as course content providers or course platform providers. *Analysts* reference the data to examine it for descriptive, inferential or predictive insights. The role of the analysts is to visualize, descriptively analyze, use machine learning or otherwise interpret some set of data within the database. Analysts extract, condition (e.g. impute missing values, de-noise), and create higher level variables for modeling and other purposes from the data. To perform this analysis, they first transform the data into the standard schema and compose *scripts* or use publicly available scripts when it suffices. They also contribute their scripts to the archive so others can use.
- It identifies two types of secondary actors: the *crowd*, and the data science experts (database experts and privacy experts). When needs arise, the community can seek the help of the *crowd* in innovative ways. Experts contribute to the community by providing state-of-the art technological tools and methods.
- A common standardized and shared schema into which the data is stored. The schema is agreed upon by the community, generalizes across platforms and preserves all the information needed for data science and analytics.
- A shared community-oriented repository of data extraction, feature engineering, and analytics scripts.
- Over time the repository and the schema, both open ended, grow.

This concept offers the following:
The benefits of standardization: The data schema standardization implies that the raw data from every course offering will be formatted the same way in its database. It ranges from simple conventions like storing event timestamps in the same format to common tables, fields in the tables, and relational links between different tables. It implies compiling a scientific version of the database schema that contains important events, fields, and dictionaries with highly structured data is amenable for scientific discovery. Standardization supports cross-platform collaborations, sharing query scripts, and the definition of variables which can be derived in exactly the same way for irrespective of which MOOC database they come from.
Concise data storage: Our proposed schema is "loss-less", i.e. no information is lost in translating raw data to it. However, the use of multiple related tables provides more efficient storage.
Savings in effort: A schema speeds up database population by eliminating the steps where a schema is designed. Investigating a dataset using one or more existing scripts helps speed up research.
Sharing of data extraction scripts: Scripts for data extraction and descriptive statistics extraction will be open source and can be shared by everyone.

Some of these scripts could be very general and widely applicable, for example: ”For every video component, provide the distribution of time spent by each student watching it?” and some would be specific for a research question, for example generation of data for Bayesian knowledge tracing on the problem responses. These scripts could be optimized by the community and updated from time to time.

Crowd source potential: Machine learning frequently involves humans identifying explanatory variables that could drive a response. Enabling the crowd to help propose variables could greatly scale the community's progress in mining MOOC data. We intentionally consider the data schema to be independent of the data itself so that people at large, when shown the schema, optional prototypical synthetic data and a problem, can posit an explanatory variable, write a script, test it with the prototypical data and submit it to an analyst. The analyst can assess the information content in the variable with regards to the problem at hand and rank and feed it back to the crowd, eventually incorporating highly rated variables into learning.

A unified description for external experts: For experts from external fields like“Very Large Databases/Big Data” or ”Data Privacy”, standardization presents data science in education as unified. This allows theme to technically assist us with techniques such as new database efficiencies or privacy protection methods.

Sharing and reproducing the results: When they publish research, analysts share the scripts by depositing them into a public archive where they are retrievable and cross-referenced to their donor and publication.

Our concept presents the following challenges:

Schema adequacy: A standardized schema must capture all the information contained in the raw data. To date, we have only verified our proposed schema serves the course we investigated. We expect the schema to significantly change as more courses and offerings are explored. It will be challenging to keep the schema open ended but not verbose. While a committee could periodically revisit the schema, a more robust approach would be to let it evolve through open access to extension definitions then selection of good extensions via adoption frequency. This would embrace the diversity and current experimental nature of MOOC science and avoid standard-based limitations. One example of a context similar to the growth of MOOCs is the growth of the internet. HTML and Web3.0 did not rein in the startling growth or diversity of world wide web components. Instead, HTML (and its successors and variants) played a key role in delivering content in a standardized way for any browser. The semantic web provides a flexible, community driven, means of standards adoption rather than completely dictating static, monolithic standards. We think there are many lessons to learn from the W3C initiative. To whit, while we provide ideas for standards below, we propose that, more importantly, there is a general means of defining standards that allow interoperability, which should arise from the examples we are proposing.

Platform Support: The community needs a website defining the standard data template and a platform assisting researchers in sharing scripts. It requires tests for validating scripts, metrics to evaluate new scripts and an repository of scripts with efficient means of indexing and retrieval.

Motivating the crowd: How can we encourage large scale script composition and sharing so the crowd will supply explanatory variables? How can we provide useful feedback when the crowd is not given the data? KAGGLE provides a framework from which we can draw inspiration, but it fundamentally differs from what we are proposing here. KAGGLE provides a problem definition, a dataset that goes along with it, whereas we are proposing that we share the schema, propose a problem, give an example of a set of indicators and the scripts that enabled their extraction, and encourage users to posit indicators and submit scripts. Such an endeavor requires us to: define metrics for evaluation of indicators/features given the problem, provide synthetic data (under the data schema) to allow the crowd to test and debug their feature engineering scripts, and possibly visualizations of the features or aggregates over their features (when possible), and most importantly a dedicated compute resource that will perform machine learning and evaluate the information content in the indicators.

3 Schema description

We surveyed a typical set of courses from Coursera and edX. We noticed three different modes in which students engage with the material. Students observe the material by accessing all types of resources. In the second mode they submit material for evaluation and feedback. This includes problem check-ins for lecture exercises, homework and exams. The third mode is in which they collaborate with each other. This includes posting on forums and editing the wiki. It could in future include more collaborative frameworks like group projects. Based on these three we divide the database schema into three different tables. We name these three modes as *observing*, *submitting* and *collaborating*. We now present the data schema for each mode capturing all the information in the raw data.

3.1 The observing mode

In this mode, students simply browse and observe a variety of resources available on the website. These include the *wiki*, *forums*, *lecture videos*, *book*, *tutorials*. Each unique resource is usually identifiable by a *URL*. We propose that data pertaining to the observing mode can be formatted in a 5-tuple table: *u_id* (*user id*), *r_id* (*resource id*), *timestamp*, *type_id*, *duration*. Each row corresponds to one click event pertaining to student. Two tables that form the dictionaries accompany this event table. The first one maps each unique *url* to *r_id* and the second one maps *type_id* to resource type, i.e., *book*, *wiki*. Splitting the tables into event and dictionary tables allows us to reduce the sizes of the tables significantly. Figure 4 shows the schema and the links.

3.2 The submitting mode

Similar to the table pertaining to the observing mode of the student, we now present a structured representation of the problem components of the course.

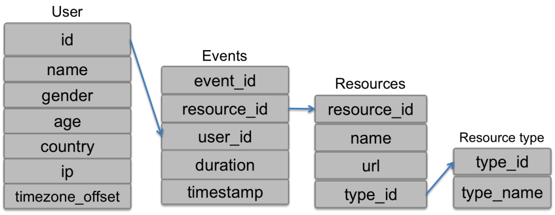

Fig. 2. Data schema for the observing mode

A typical MOOC consists of assignments, exams, quizzes, exercises in between lectures, labs (for engineering and computer science). Unlike campus based education, students are allowed to submit answers and check them multiple times. Questions can be multiple choice or a student can submit an analytical answer or even a program or an essay. Assessments are done by computer or by peers to evaluate the submissions [1]. We propose the following components:

Submissions table: In this table each submission made by a student is recorded. The 5 tuple recorded is *u_id*, *p_id*, timestamp, the answer, and the attempt number.

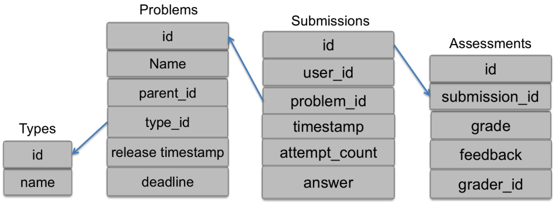

Fig. 3. Data schema for the submitting mode.

Assessments table: To allow for multiple assessments this table is created separately from the submissions table. In this table each assessment for each submission is stored as a separate row. This separate table allows us to reduce the size since we do not repeat the *u_id* and *p_id* for each assessment.

Problems table: This table stores the information about the problems. We *id* the smallest problem in the entire course. The second field provides the name for the problem. The problem is identified if it is a sub problem within another problem by having a parent *id*. Parent *id* is a reflective field in that its entries are one of the problem *id* itself. Problem type *id* stores the information about whether it is a homework, exercise, midterm or final. The table also stores the problem release date and the problem submission deadline date as two fields. Another table stores the id for problem types.

3.3 The Collaborating mode

Student interact and collaborate among themselves throughout the course duration through forums and wiki. In forums a student either initiates a new thread or responds to an existing thread. Additionally students can up vote, and down vote the answers from other students. In wiki students edit, add, delete and initiate a new topic. To capture this data we form the following tables with the following fields:

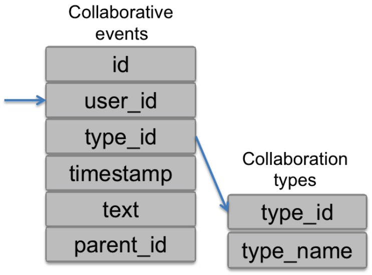

Fig. 4. Data schema for collaborating mode

Collaborations table: In this table each attempt made by a student to collaborate is given an *id*. The 5 fields in this table are *u_id*, collaboration type (whether wiki or forum), timestamp, the pointer to the text inserted by this user, and the parent *id*. The last field is a reflective field as well.

Collaboration type table: In this table the collaboration type *id* is identified with a name as to whether it is a wiki or a forum.

4 The edX 6.002x case study

edX offered its first course *6.002x: Circuits and Electronics* in the Fall of 2012. 6.002x had 154,763 registrants. Of these, 69,221 people looked at the first problem set, and 26,349 earned at least one point on it. 13,569 people looked at the midterm while it was still open, 10,547 people got at least one point on the midterm, and 9,318 people got a passing score on the midterm. 10,262 people looked at the final exam while it was still open, 8,240 people got at least one point on the final exam, and 5,800 people got a passing score on the final exam. Finally, after completing 14 weeks of study, 7,157 people earned the first certificate awarded by MITx, showing that they successfully completed 6.002x.

The data corresponding to the behavior of the students was stored in multiple different formats and was provided to us. These original data pertaining to the observing mode was stored in files and when we transcribed in the database with fields corresponding to the names in the "*name-value*" it was about the size of around 70 GB. We imported the data into a database with the schema we described in the previous subsections. The import scripts we had to build fell into two main categories:

- reference generators, which build tables listing every user, resource and problem that were mentioned in the original data.
- table populators, which populate different tables by finding the right information and converting it if needed.

The sizes and the format of the resulting tables is as follows: submissions: 341 MB (6,313,050 rows); events: 6,120 MB (132,286,335 rows); problems: 0.08 MB; resources: 0.4 MB; resource types: 0.001 MB; users: 3MB. We therefore reduced the original data size by a factor of 10 while keeping most of the information. This allows us to retrieve easily and quickly information on the students' activities. For example, if we need to know what is the average number of pages in the book a student read, it would be around 10 times faster. Also, the relative small size of the tables in this format allows us to do all the work in memory on any relatively recent desktop computer. For more details about the analytics we performed as well as the entire database schema we refer the reader to [2] [2]

5 Conclusions and future work

In this paper, we proposed a standardized data schema and believe that this would be a powerful enabler for ours and others researchers involved in MOOC data science research. Currently, we after building databases based on this schema we are developing a number of analytic scripts that extract multiple attributes for a course. We intend to release them in the near future. We believe it is timely to envision an open data schema for MOOC data science research.

Finally, we propose that as a community we should come up with a shared standard set of features that could be extracted across courses and across platforms. The schema facilities sharing and re-use of scripts. We call this the "feature foundry". In the short term we propose that this list is an open, living handbook available in a shared mode to allow addition and modification. It can be implemented as a google doc modified by the MOOC community. At the moocshop we would like to start synthesizing a more comprehensive set of features and developing the handbook. Feature engineering is a complex, human intuition driven endeavor and building this handbook and evolving this over years will be particularly helpful.

References

1. Piech, C., Huang, J., Chen, Z., Do, C., Ng, A., Koller, D.: Tuned models of peer assessment in MOOCs. In: Proceedings of The 6th International Conference on Educational Data Mining (EDM 2013). (2013)
2. Dernoncourt, F., Veeramachaneni, K., Taylor, C., O'Reilly, U.M.: Methods and tools for analysis of data from MOOCs: edx 6.002x case study. In: Technical Report, MIT. (2013)

[2] For the full MOOCdb database schema, see `http://bit.ly/MOOCdb`)