Hutt Stephen (Orcid ID: 0000-0002-7041-7472)

Controlled Outputs, Full Data: A Privacy-Protecting Infrastructure for MOOC Data

Stephen Hutt
0000-0002-7041-7472
University of Pennsylvania

Ryan S. Baker
0000-0002-3051-3232
University of Pennsylvania

Michael Mogessie Ashenafi
0000-0001-6769-5941
Carnegie Mellon University

Juan Miguel Andres-Bray
0000-0001-7126-7819
University of Pennsylvania

Christopher Brooks
0000-0003-0875-0204
University of Michigan

Correspondence Address: Stephen Hutt,
3700 Walnut Street,
Philadelphia, PA,
19104

Email: hutts@upenn.edu

Acknowledgements

This research was supported by the National Science Foundation (NSF) (NSF-OAC#1931419). Any opinions, findings, and conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the NSF.

Conflict of Interest Statement

No conflict of interest (financial or non-financial) has been declared by the authors.

This is the author manuscript accepted for publication and has undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process, which may lead to differences between this version and the Version of Record. Please cite this article as doi: 10.1111/bjet.13231

ABSTRACT

Learning analytics research presents challenges for researchers embracing the principles of open science. Protecting student privacy is paramount, but progress in increasing scientific understanding and improving educational outcomes depends upon open, scalable, and replicable research. Findings have repeatedly been shown to be contextually dependent on personal and demographic variables, so how can we use this data in a manner that is ethical and secure for all involved? This paper presents ongoing work on the MOOC Replication Framework (MORF), a big data repository and analysis environment for Massive Open Online Courses (MOOCs). We discuss MORF's approach to protecting student privacy, which allows researchers to use data without having direct access. Through an open API, documentation, and tightly controlled outputs, this framework provides researchers with the opportunity to perform secure, scalable research and facilitates collaboration, replication, and novel research. We also highlight ways in which MORF represents a solution template to issues surrounding privacy and security in the age of big data in education and key challenges still to be tackled.

1 Introduction

All too often, positive values conflict with each other. A debate of this nature has emerged in the recent years in the discourse around learning analytics. On the one side of this debate is the value of beneficence (Prinsloo & Slade, 2017), the idea that learning analytics can enable interventions which improve the chance that students will succeed or complete a course, ultimately benefiting their lives. Models and algorithms in this area can be enhanced through open science (van der Zee & Reich, 2018). Open science makes it possible to replicate and verify models (Echtler & Häußler, 2018), debate around how they function (Chakraborty et al., 2018; Chatzimparmpas, Martins, Jusufi, & Kerren, 2020), and compete to see which approaches are most effective (Foster & Deardorff, 2017).

On the other side of this debate is the value of nonmaleficence (Corrin et al., 2019) – in this context, the idea that the pursuit of learning analytics should not harm learners. One major nonmaleficence concern is around privacy. Learners trust platforms with highly personal data, the disclosure of which can cause harm. This has led to several developments guiding the use of educational data, varying in different countries, and tracking developments in privacy in society and policy more broadly. For instance, in the European Union, it has led to the right to be forgotten, mandating that organizations be required to delete individual data upon request (Politou, Alepis, & Patsakis, 2018). In the United States, it has led to increasing pressure on commercial organizations handling learner data to delete all identifying information at the end of every year (Consortium & others, 2018). In general, this trend has led to greater concern about how educational data is handled and a strong emphasis on avoiding disclosure of personally identifying information.

This challenge can be particularly difficult for learning data from online courses, including massive online open courses (MOOCs). It is not sufficient for these courses to simply remove user names from the data and declare the data deidentified. Data from these courses can often contain personal details about a student, from their demographics and IP address, to forum posts, responses, and essays which sometimes contain subtly identifying information (see, for example, Frankowski, Cosley, Sen, Terveen, & Riedl, 2006). Extensive usage data can be used for several beneficent purposes -- improving the quality of algorithms (Gardner & Brooks, 2018), checking for and removing algorithmic bias (Kizilcec & Lee, 2020) -- but is hard to fully deidentify.

In this article, we present a potential solution to address both the goals of beneficence (through open science) and nonmaleficence (through privacy protection) in a research architecture for massive online open course data. The goal is to respect privacy but promote open, scalable, replicable research that can stand up to scrutiny, and ultimately benefit learners. In what follows, we detail the ongoing development of the MOOC Replication Framework (MORF), a framework to promote Ethical and Trustworthy

Learning Analytics that supports the goals of learners and researchers alike. MORF allows researchers to perform computation on a growing dataset without ever having direct access to the data, protecting student privacy. Our continuing development aims to address current challenges in learning analytics whilst making data more accessible to the research community.

1.1 The Trouble With Personal Identifying Information

Personal Identifying Information (PII) is defined as "Any representation of information that permits the identity of an individual to whom the information applies to be reasonably inferred by either direct or indirect means" (U.S. Department of Labor, n.d.). In education, PII can take several forms. Classic examples such as name, date of birth, and email address are immediately obviously PII. However, PII can also be contained in data such as discussion forum responses, where a learner may use their name or nickname, discuss their hometown or job, or identify themselves in other ways that can be mapped back to them as an individual ("Hi! My name is Ryan. I live in Pennsylvania and I'm a professor at the University of Pennsylvania! I work on learning analytics, and I'm a huge fan of ten-pin bowling.."). Even variables that we might typically aggregate across, such as gender or race, can become identifying in certain circumstances (e.g., a student's identity is underrepresented in the class/group).

1.1.1 Important Uses of PII in Educational Research and Practice

A student's identity is critical to their learning experiences, and can influence how and when the student succeeds, as well as what supports most benefit them. Culturally responsive pedagogy has increasingly provided evidence that instructional methods are more effective if they take students' identity and individual lived experiences into account (Howard & Terry Sr, 2011). As a range of studies have now shown, specific online interventions may have different effectiveness for different groups of students (Arroyo, Burleson, Tai, Muldner, & Woolf, 2013; Finkelstein, Yarzebinski, Vaughn, Ogan, & Cassell, 2013; Kizilcec, Pérez-Sanagustin, & Maldonado, 2017). Identity has many dimensions, including (but not limited to) race, class, gender expression, sexual orientation, ethnicity, religion, nationality, age, language, and ability – all considered PII. In addition, the intersectional relationships between these variables play a major role in how identity manifests and influences learning (Ro & Loya, 2015). We cannot determine which interventions work for which students – or be confident that an intervention works for the full diversity of students – without the use of PII.

Another key use for PII is in the study and evaluation of algorithmic bias – the concept that algorithms often encode the biases of their developers or the surrounding society, producing predictions or inferences that are clearly discriminatory towards specific groups (Baker & Hawn, 2021) . These problems have been consistently documented in education, with bias in testing, for example, being discussed since the 1960s. In order to assess and attempt to remove this bias, it is necessary to have access to data that identifies each student's group memberships. This data is invariably PII. As Baker & Hawn (2021) argue, we cannot achieve fairness without knowing whether an algorithm is biased, and we cannot determine that without PII. Identifying and fixing algorithmic biases hinges upon the use of PII and cannot be completed without it.

1.1.2 Data Sharing and Researcher Ecosystem

In education research, data sharing often poses challenges. Data is typically collected in partnership with educators, administrators, and students, who authorize the collection of data for a specific study/set of research questions, and often actively prohibit the distribution of data to third parties. Data can be deidentified, but given how intrinsically personal educational data can be, this task can be labor-intensive. Worse, some of the easier forms of deidentification (such as removing all forum post data prior to sharing (EdX, 2020) lead to data no longer being useable for a wide range of research and development goals.

Sharing data on a by-request basis (e.g., Wolins, 1962) and carefully crafting data agreements has long been a potential solution, but is often ineffective. For example, (Wicherts, Borsboom, Kats, & Molenaar, 2006) contacted owners of 249 datasets, only receiving a response from 25.7%., a response rate similar to that noted in (Wolins, 1962) following requesting data from 37 APA articles (though many years earlier and prior to email). The task of sharing data requires a time investment from researchers, typically with no incentive. Moreover, the process can be stalled by email address or institution changes.

Restrictions on data sharing pose particular issues in terms of the increasing move towards Open Education Science (van der Zee & Reich, 2018), a subfield of Open Science (see Fecher & Friesike, 2014). This movement seeks to address problems of transparency and access, specifically in education research, addressing issues of publication bias, lack of access to original published research, and the failure to replicate. The practices proposed by Open Education Science fall into four categories, each related to a phase in the process of educational research: 1) open design, 2) open data, 3) open analysis, and 4) open publication.

Of most relevance to the current work are Open Data and Open Analysis. The principle of Open Data aims to make data and other research materials freely accessible on public repositories for the purposes of replication, evaluation, and scrutiny. As noted above, this goal often presents challenges for educational data in cases where it was originally agreed that data not be shared, or where PII issues limit what can be shared. Open Analysis poses that analysis and methods should be able to be systematically reproduced, a goal often accomplished through code sharing. In these cases, source code used for analysis is often made publicly available through online repositories such as GitHub or preregistration sites. The challenge here is, beyond initial examination, this code is typically not useful without the data needed to conduct the analysis, meaning that without Open Data, Open Analysis is typically not possible. In addition, issues of code rot and dependency hell (Boettiger, 2015) often make it impossible to run code once the libraries the code depends on have changed.

1.2 Our Approach

Our approach is an infrastructure that protects student privacy while recognizing the value of PII to education research and the importance of open, replicable analysis. MORF gives users access to extensive learner data from MOOCs, allowing researchers to conduct analysis on unrestricted, complete data, while preventing them from directly viewing the data. MORF is currently installed on remote cloud servers rented by the University of Pennsylvania and the University of Michigan (version 2.0), and is designed so that any institution wishing to have its own implementation can relatively straightforwardly install it on their own servers. Though the MORF infrastructure could be used with other forms of data, the infrastructure has focused on MOOC data, to begin with, and currently, all data ingestion is focused on MOOC platforms such as EdX and Coursera. MOOC data presents a convenient starting point as we establish this privacy-protecting framework. MOOCs are self-contained courses, with large volumes of data, and already defined data structures. There is also a growing body of MOOC scholarship, with increasing concerns about replicability, that MORF can support.

MORF gives users the freedom to dictate a study's overall design – from feature extraction to model evaluation. In order to protect the data available, users are provided a sample dataset and a minimum working job example. Both resources give users the ability to design their study and ensure that it will run on MORF. Once a job is submitted to MORF (via a public URL), the user's code is executed against the framework's database to extract features, which are then used to train and test predictive models according to the user's research design. All intermediate outputs used in these processes are outputted and stored privately on remote Amazon Web Services (AWS) servers. Finally, once the job is complete, the model's evaluation metrics are sent to the user's email. By preventing viewing of personally identifying information, and offering access to a restricted set of evaluation metrics, MORF allows extensive analysis but protects student privacy.

2 PRIOR WORK

In this section, we review past work relevant to the open science challenges that MORF attempts to address. Due to the expanding interest in this space, we focus our review on work conducted in educational domain, for brevity.

2.1 Post-hoc analysis platforms

The first large source of publicly-available educational interaction data was the Pittsburgh Science of Learning Center Datashop, now called Learnsphere (Koedinger et al., 2010; Stamper et al., 2016). DataShop contains (and contained) data from hundreds of thousands of students using dozens of different learning platforms, with a particular focus on intelligent tutoring system data. The data in DataShop was primarily in the form of interaction logs – semantically meaningful actions made by the student within the learning system such as entering an answer or requesting a hint. Other data, such as test data, typically is included as separate data files available for download.

DataShop was originally conceived with a model where a small number of analyses would occur online – this functionality was designed primarily for educational researchers without a data science background. For more advanced analyses involving data mining and machine learning, DataShop enabled researchers to download a version of the data with all student identifiers removed. DataShop data has been used in this fashion by hundreds of data mining researchers. Later work with the Tigris infrastructure enables researchers to specify and run more complex analyses online, using a drag-and-drop graphical user interface (Liu, Koedinger, Stamper, & Pavlik, 2017). Researchers can add models and modeling tools to Tigris, and a small number of tools have been added by external researchers (e.g.Paquette, Baker, & Moskal, 2018).

The moocDB database schema was proposed and developed to standardize the vast amounts of data generated by multiple MOOC platforms. It has received considerable discussion within MOOC scholarship as a way of making data compatible across MOOC platforms (e.g., Baker & Inventado, 2014; Pournaras, 2017; Sun et al., 2019), but was rarely used except in studies involving its developers (Han, Veeramachaneni, & O'Reilly, 2013). Its last published use was in 2014 (Han & others, 2014).

MoocRP is an analytics tool that was developed with a goal of supporting replicable research using MOOC data (Pardos & Kao, 2015). moocRP allows for the implementation of several analytic models, with the goal of facilitating the re-use and replication of an analysis in a new MOOC. One of its key features was its ability to display visualizations. For example, in its first published case study, moocRP was used to employ Bayesian Knowledge Tracing (Pardos & Heffernan, 2010) to assess current and previous knowledge among learners in a MOOC (Pardos & Kao, 2015), producing visualizations which an instructor could use to determine who in their class is lacking the requisite prior knowledge to succeed. However, moocRP did not scale beyond analyses of single MOOCs, thus not facilitating the types of broad, cross-contextual research that are needed to get MOOC research past its own replication crisis. moocRP did not achieve widespread use, and its source code and documentation have not been updated since 2016.

One educational platform that has been active in sharing its data is ASSISTments (Heffernan & Heffernan, 2014), which has released data sets publicly on its webpage[1] for download since 2010 (e.g. (Selent, Patikorn, & Heffernan, 2016)). ASSISTments is an online grant-funded mathematics learning platform, provided to teachers and students around the world as a free public service of Worcester Polytechnic Institute (Ostrow & Heffernan, 2016). Today, researchers can gain access to data either via the ASSISTments website or for specific studies, through the Assessment of Learning Infrastructure (ALI). ALI provides automated reports that provide basic analyses and pre-processed CSV files featuring

[1] https://sites.google.com/site/assistmentsdata/

the raw data logged by ASSISTments as students work through an experimental assignment (Ostrow & Heffernan, 2016). These data sets, like the DataShop data, are primarily in the form of interaction logs. Other data, such as test data, longitudinal outcome data, and affect observation data, is also included as separate data files available for download. ASSISTments data sets have been used by hundreds of external researchers, including in a longitudinal data competition (Patikorn, Heffernan, & Baker, 2018). ASSISTments attempts to fully deidentify data, removing both student identifiers and student responses that reveal PII. Researchers are also required to agree (through terms of service) not to reidentify students. Should the researcher require more in depth access, individual data agreements must be completed with the ASSISTments foundation. Thus, this data and experimentation infrastructure provides a very rich (and far reaching) insight into student learning, but does not on its own facilitate certain research questions. For example, researchers can examine the effect of an question or feedback mechanism, but not individual differences (that rely on demographics) without additional data agreements and access.

In aggregate, these platforms provide access to data for secondary analysis, but they face challenges in resolving the dilemma of balancing between privacy/security and the ability to conduct the full range of analyses that could be conducted with this data. While in general these platforms share data openly they do so by removing identifiers in order to reduce privacy risks. By sharing limited data, many important analyses (such as longitudinal analyses and algorithmic bias analyses) cannot be conducted using this data. At the same time, by sharing data openly, it is impossible to prevent all possible reidentification attacks (see, for instance, Yacobson, Fuhrman, Hershkovitz, & Alexandron, 2021).

2.2 Experimentation Platforms

In addition to infrastructures for data analysis, there has also been work providing an infrastructure for large scale data collection. For example, Ed-Tech Research Infrastructure to Advance Learning Science (E-TRIALS – an additional layer to the ASSISTments system described above) facilitates Randomized Control Trials and A/B testing with real student populations. Researchers then have access to this data through the mechanisms described above. By leveraging the userbase of the ASSISTments platform, E-TRIALS allows researchers to access larger volumes of data than may be collected otherwise, while still protecting student privacy. As with general ASSISTments data, demographic and individual differences data is not shared unless additional agreements are in place.

E-TRIALS incorporates open access content from ASSISTments and wraps this with an experimental platform, the Teracotta framework[2], which is provided as an open-source IMS LTI plugin suitable for embedding within traditional learning management systems. The goal of Teracotta is to support context-driven experimentation both through large-scale multi-institution experiments (e.g., Fyfe et al., 2021) as well as experiments within a single course. While the framework is still under development, it aims to ease much of the logistical burden of running robust and ethical experiments in education, including assignment of learners to different conditions (which may include crossover designs), collection of and blinding to the instructor informed consent, and connection between institutional units.

2.3 Public Datasets

Beyond the infrastructures and platforms described above, there have also been efforts to provide datasets for public download. For example, the Open University Learning Analytics Dataset (OULAD)[3] was published in 2015 through the UC Irvine Machine Learning Repository. The dataset contains deidentified data from seven online courses and accompanying documentation for the explanation of variables. The data does contain data for some individual differences (gender, age, etc.) and can be used for a number of machine learning analysis for online courses. Several data competitions, such as the 2010 KDD Cup[4], the

[2] https://terracotta.education/

[3] https://archive.ics.uci.edu/ml/datasets/Open+University+Learning+Analytics+dataset

[4] https://pslcdatashop.web.cmu.edu/KDDCup/downloads.jsp

NAEP Data Mining Competition 2019, and the NeurIPS 2020 Education Challenge, have also made data available. These datasets vary in content and context, and are typically one-off datasets as opposed to a commitment of continuously updated data. As with the more comprehensive solutions to data sharing seen in the PSLC DataShop and ASSISTments system, these datasets typically offer partially redacted data (less so with OULAD than other data sets) and may still present some limited privacy risk – with the degree of data redaction corresponding inversely to the degree of privacy risk.

3 Our Solution – The MOOC Replication Framework

3.1 MORF 1.0

In its first iteration (Andres et al., 2018), MORF ran on a dedicated server and was intended to replicate findings that could be posed as human-understandable if-then rules such as "If a student who is <attribute> does <operator>, then <outcome: completes or does not complete>", where an attribute is a piece of information about a student and an operator is an action within a given MOOC. MORF returned whether a finding was statistically significant in a new course (or set of courses), and the effect size of the relationship. In MORF 1.0, the only outcome that could be assessed was MOOC completion. This initial implementation was developed using a production-system framework (Jess). While this allowed for the examination of many published findings (Andres et al., 2018; Andres, Baker, Siemens, Gašević, & Spann, 2017), it represented a limited subset of MOOC scholarship.

3.2 MORF 2.0

The second iteration of MORF (MORF 2.0) built upon the success of 1.0 for replication studies and added increased functionality for studies and findings that could not be distilled to if-then rules. Through updates to the framework, we developed a new predictive modeling module and easier access to MOOC data, enabling more direct forms of replication research. Using the predictive modeling research cycle (extract, train, test, and evaluate), users could first program their own feature extraction scripts and specify the outcome of interest (completion or dropout). Users could also provide high-level experimental workflows, such as how model training and testing should occur and whether cross-validation or a holdout set should be used. The remaining steps of the cycle were then handled by MORF, with a preprogrammed set of classification and evaluation algorithms.

MORF 2.0 ran on a dedicated server and consisted of two main components: an open-source Python API for specifying the workflow of an experiment (the "MORF API"), and a Platform-as-a-Service, which was a running instance of MORF's backend infrastructure coupled with computational resources and a large MOOC dataset. In order to use this version of MORF, users needed to create and submit configuration files to MORF, either using an HTTP request or using a MORF API function. These configuration files contained human-readable job metadata, including a pointer to an executable Docker image which contained all code, software, and dependencies for the experiment. Once the job was done executing on MORF, the analysis results were then emailed to the user.

3.3 MORF 2.1

The latest version of MORF (MORF 2.1, henceforth referred to just as MORF) offers additional flexibility in study design. In addition to programming their own feature extraction scripts, users are now able to implement their own analyses beyond just predictive modeling, and output using a range of pre-approved libraries. In this version of MORF, research artifacts (e.g., docker file, extraction code, analysis code, etc.) are still sent to the platform using the MORF API. As with MORF 2.0, the job is conducted within the MORF platform, and all final outputs are sent to the user via email.

Beyond these changes, the latest version of MORF has been redesigned to address issues regarding usability, scalability, and security. In earlier versions, MOOC course data was stored in flat database

dumps that were not easily accessible by users without considerable expertise. Users still did not have direct access to files, thus in order to use the data, users had to design their job to first manipulate the data into a usable form before conducting analysis. Each MORF job would typically establish a temporary database and load data into it for use. These databases would then be destroyed when the job was complete. While many users may be adept at querying a database, establishing a database with the appropriate relationships can be challenging, especially without direct access to data for debugging purposes. This necessity also led to significant CPU and memory overhead on the MORF sever that needed to be repeated in projects using the same data. This approach did not scale efficiently as the number of jobs submitted by users increased. To address these issues and take advantage of the security and scalability features provided by cloud services, MORF is now deployed on AWS Cloud. AWS is an on-demand cloud computing service that allows for varying computation and storage requirements.

MORF 2.1 is currently installed and maintained by a team of MORF Administrators (further described below) at the University of Pennsylvania. This team maintains the data storage and broader MORF infrastructure (more detail below). Only members of the MORF Administrators team have direct access to the data used in the MORF infrastructure. Users of MORF do not need to be affiliated with the University of Pennsylvania, but must register in order to obtain appropriate credentials (an API Key, more detail below).

3.3.1 MORF Architecture

The MORF backend provides users an indirect, unrestricted and read-only access to MOOC data for predictive modeling. However, the information users can directly access from MORF is restricted. That is, a user's code may access the data only through running code on MORF. All code is run in an isolated environment. The benefits of this are twofold: First, environments can be built with no restriction on programming language or software used for analysis. Second, access to resources such as networking, databases, and file systems can be restricted when running user code in an isolated environment. Output from the isolated environment is then passed to an output pipeline that evaluates model performance. Functions permitted within this approved pipeline are the only output that users have access to.

MORF is implemented in AWS as a service that utilizes several cloud services. The MORF backend is a RESTFul web service that runs inside an EC2 virtual machine instance that is accessible through a load balancer. The load balancer is a single point of entry for incoming traffic that distributes requests to multiple instances of MORF based on metrics such as memory and CPU usage. The load balancer also serves as a firewall by restricting access to the MORF backend according to security rules, such as restricting access to certain IP addresses. Using a load balancer is one of the common ways to automatically scale resources and adjust to demand. Rather than requiring each job to load data into a database, MOOC data is pre-imported and stored in RDS MySQL databases. These databases are accessible through a proxy that MORF jobs can only access from within the isolated environment. MORF also uses a queueing service in AWS to queue and process jobs. After a job is processed, results are sent to the user via email through an email service provided by AWS. Figure 1 shows the MORF architecture and the AWS services it utilizes.

Secure access to MORF on AWS is provided through firewall security rules that prevent direct access to the data. The server that runs MORF on AWS can only be accessed through a load balancer and does not accept direct requests from the internet. Moreover, user jobs are run in an environment that does not allow outbound connections to the internet. Additional security layers are provided by IT Administrators (in our case, the University of Pennsylvania IT department), who manage the AWS Account and provide regular security audits of our infrastructure.

In the current design, this architecture would be repeated for each university that wished to run an installation of MORF. In discussions with university leadership and IT teams, we learned that it is important for universities to be able to control their own separate cloud servers rather than sending it to a single server that controls all universities' data. Each university controlling their own server makes it possible for their IT department to exercise due diligence over their own security, as well as their students' data.

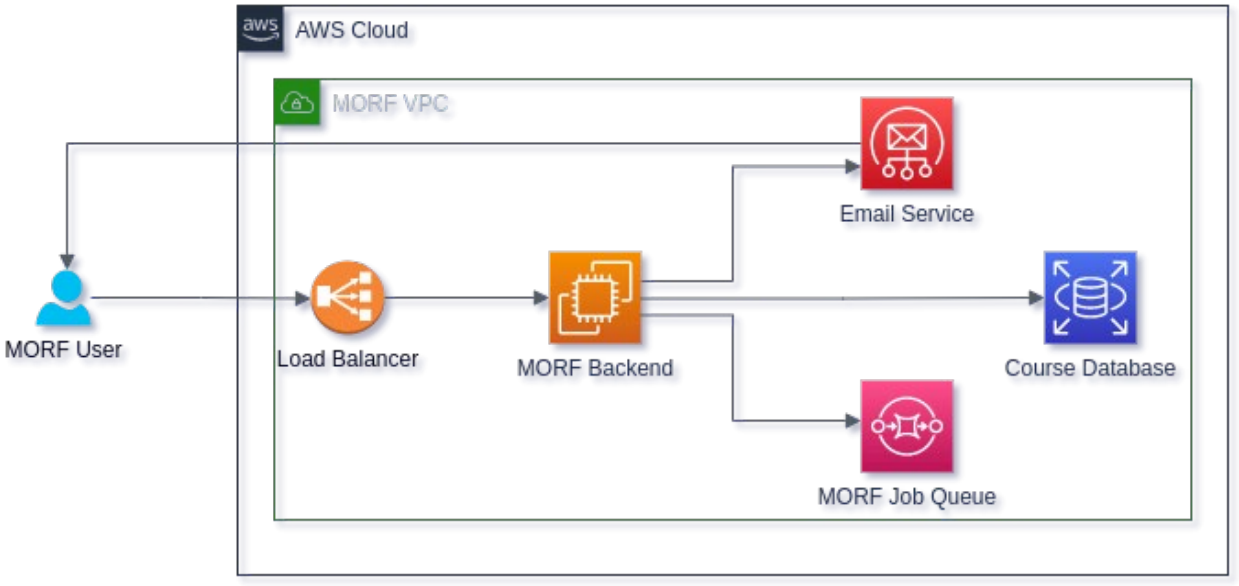

Figure 1. Diagram of the MORF Architecture

3.3.2 MORF Workflow

The MORF workflow starts with users utilizing the MORF API to submit a job to the MORF backend. A job request includes an API key (linking a user to the job), a URL to the MORF backend, and a job payload. The payload contains instructions on how the MORF backend should initialize the isolated environment and any associated files to be copied and executed in the isolation environment. Concretely, this isolation environment is a Docker container, preserving library versions and the runnable environment as well as the code.

After a job is successfully submitted to the MORF backend, it is placed in a queue. MORF then retrieves the job from the queue and builds the specified image which is then used to launch the isolated environment (a container). It then runs the user code within that container. MORF users are instructed to write output to a volume that is attached to the container for processing as described above.

Job execution is logged by MORF. If there are any errors executing a job, the user receives an automated email that informs them that the job was not successful. The MORF technical team will also receive an automated email with detailed error messages that can be provided to the user so long as there is no leaked PII within the error messages. We use the Simple Email Service (SES) provided by AWS to email users the outcome of their job submissions, such as successful completion and if the job was not completed due to an error. If a job completes successfully, the email also contains the outputs of the model's performance. Figure 2 shows an overview of a MORF job workflow.

Any and all jobs that have access to the MORF database must follow this workflow. However, this can prove challenging for the initial development of code and debugging (see limitations). For this purpose, we have some examples of both SQL queries and sample placeholder data for users to test/debug their code offline. Sample data is available upon request – when a new type of sample data is needed, it is created by the MORF Administrators.

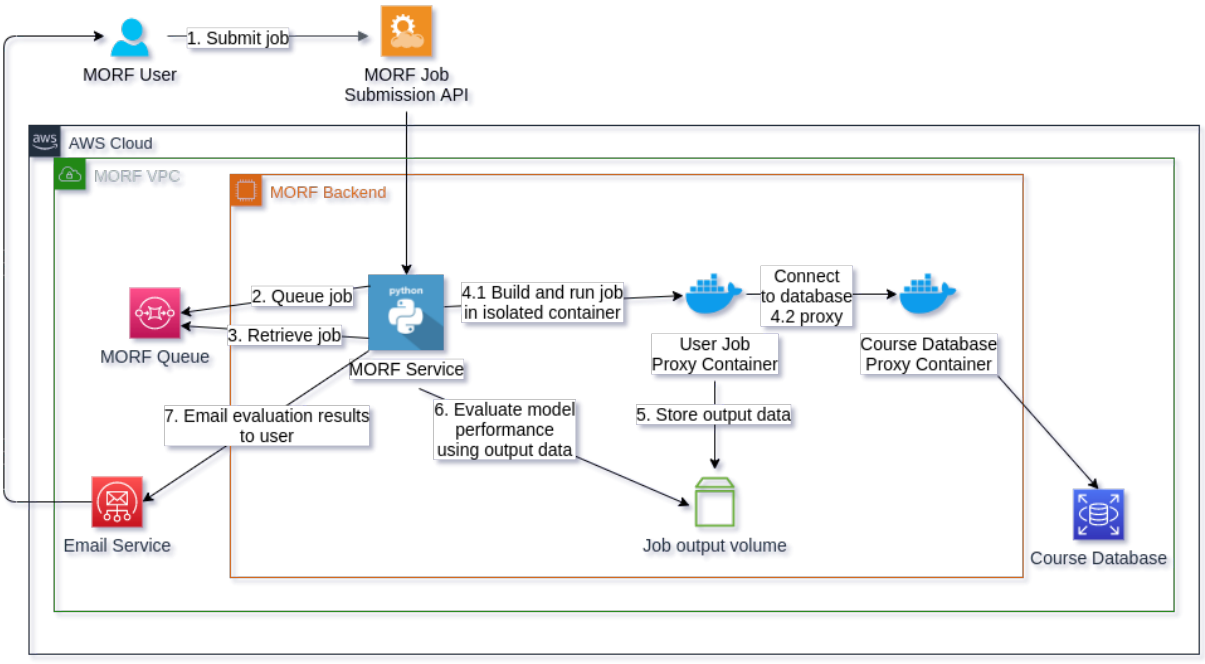

Figure 2. Diagram showing MORF Job Workflow

3.3.3 MORF Users

The MORF infrastructure has three broad groups of users: (1) Users (2) MORF Administrators, and (3) IT Administrators. We refer to Users as those running analysis within MORF. They must authenticate their interaction with an API key (for both support and security purposes), submit jobs via the MORF Job Submission API, and ultimately receive their results via email. MORF Users have no direct access to data; their access is via the API. MORF Administrators are the team that processes data received from MOOC providers (e.g., Coursera) and imports such data into the MORF databases. MORF Administrators also handle the distribution of API keys and general user management. They are also on hand for any support needs for users (e.g., code debugging, or assessing data needs). IT administrators (in our case, the IT department at the University) manage the AWS account and billing as well as provide security audits and recommendations.

Users of the MOOCs, such as instructors, or students enrolled in courses, are not considered MORF users as they do not have any interaction with the system; their data is simply imported into MORF. In some cases, Instructors may wish to use MORF for analysis, in which case they would need to register as a MORF User.

3.3.4 MORF Data

MORF 2.1 is currently used to give users execute-only access to MOOC data from 45 courses offered in English by the University of Pennsylvania from 2012 to 2015. We are currently expanding the data contained within MORF 2.1 to include data from 2016 to the present. At this time, version 2.1 of MORF is only used by a single university, the University of Pennsylvania; other universities continue to use 2.0. Of these courses, 27 had multiple sessions, resulting in a total of 98 sessions' worth of data. For each session, the following types of data are available:

a. Discussion forum posts: this SQL file contains all the data related to the threads, posts, and comments on the session's discussion forum.
b. Course data: this SQL file contains semantic-level data on every other part of the MOOC beyond discussion forum posts, such as data and metadata on student access to lecture videos and hand-in of assignments, grades, etc.
c. Clickstream: this data file contains all learner clicks within the system, where each click is logged as a JSON object.

The data imported is provided in the same format used by the MOOC provider (e.g., Coursera, EdX). Both EdX and Coursera provide scripts to import data exported from their platforms into other databases. Using these scripts preserves existing data structure, meaning that researchers may leverage existing database documentation.

3.3.5 MORF security

MORF implements multiple security features to authorize access and shield data:

a. User communication with the MORF backend is encrypted with SSL.
b. Users must be issued 4096-bit RSA-encrypted API keys before they can submit jobs.
c. The MORF backend EC2 instance can only be accessed via a load balancer. It cannot be reached directly from the internet.
d. The MORF backend EC2 instance cannot make any outbound connections.
e. A user's job is isolated inside a Docker container. The docker container cannot make any outbound connections and can only communicate with the course database proxy container.
f. Course databases themselves are not directly accessible by users.
g. The RDS database system cannot be reached from the internet.
h. Course databases are read-only.
i. If an error occurs during job processing, users receive only basic error messages that indicate at what stage the error occurred. Users will not receive any error messages that may reveal sensitive information stored in the databases. For instance, users will receive an error message that may tell them the error occurred while unzipping the job payload or that will specifically tell them the error is in their code. However, the error message will not include the runtime error that occurred while executing their code. Such messages and other system-level errors are only sent to the MORF technical team.

3.3.6 Multi-Institution Analyses within MORF

As of this writing, MORF version 2.1 is currently only in active use by the University of Pennsylvania, with active plans to upgrade the University of Michigan's infrastructure from 2.0 to 2.1. The general model for the use of MORF by multiple institutions is that each institution runs its own instance. Researchers outside the institution can run analyses according to the same processes as researchers internal to the institution. MORF can be used in analyses of multiple institutions' data by running a job

separately at each university and then exchanging intermediate values or models between institutions (but not data). For instance, a researcher could run the same research process predicting dropout in a single course at both universities, obtain metrics for each course at each university, and then combine or compare the metrics across universities. To give another example, a researcher could run an algorithm to predict student dropout in a course from one university, and then export the model (subject to verification that no fields involving PII were utilized in the model) from one university and re-import it through a Docker container to the other university.

What MORF currently does not support is the actual linking of data between institutions at the level of individual students. This means, for instance, that we cannot currently track students between MOOCs if they take some from the University of Pennsylvania and others from the University of Michigan. The use cases involving tracking students between different universities' MOOCs has not yet necessitated developing functionality for this, but this use case may become more relevant in the future if student information systems or learning management systems used at the K-12 level adopt MORF. If that happens, linking data between institutions would enable powerful longitudinal analyses. Such a step would involve not just technical challenges (setting up secure data tunnels between different institutions' installations of MORF) but also challenges involving developing legal agreements for data sharing and/or linkage that would not violate institutions' legal responsibilities around student data privacy.

3.3.7 User Experience and Technical Requirements

MORF 2.1 has been designed to reduce the technical skills and manual operations required by users to set up and submit jobs (compared to 2.0). It is our intention that the technical knowledge be limited to a standard data scientist toolset. The following technical capabilities will help users utilize MORF effectively.

a. Elementary knowledge of JSON and Python to use the MORF job submission API.
b. Basic to moderate knowledge of Docker to create the job set up and running instructions that MORF will use to run their code.
c. Intermediate knowledge of SQL and, in some cases, specific MySQL syntaxes, to execute SELECT operations on course data stored in MySQL databases.
d. Sufficient knowledge of a programming language to conduct analysis. Because jobs are run inside a Docker container, MORF is agnostic to the programming language or any software package that is used within the container.

By using a publicly available python library for job submission, getting set up to use MORF uses existing installation protocols (such as pip or anaconda) that users may already be familiar with and which have extensive support. Currently, MORF jobs are primarily submitted through the command line, but there is facility for graphical output, in the results emailed to the user. Users can also use IDEs and notebook environments (e.g., Jupyter) to develop their jobs.

In order to help users with the setup and submission of jobs, the MORF repository contains documentation and a minimum working example (MWE) that users can build upon. The MWE details two parts of MORF analysis, (1) designing an analysis, and (2) submitting the job to MORF. Once the MWE has been successfully completed, users are then able to design an analysis that is as simple or as complex as they choose.

4 Research Enabled

Though version 2.1 has only recently launched, previous iterations of MORF (1.0 and 2.0) have enabled a number of research studies. These can be split broadly into two categories, replication, and novel research. As the name suggests, MORF was initially designed to provide a data sharing platform to enable

replication studies. For example, MORF was used to conduct a replication analysis in MOOCS, investigating 21 previously published findings on learner attributes (as self-reported on pre-course surveys) and course interaction, discussion forum posting behavior, and discussion forum post sophistication (Andres et al., 2017). The study found that nine of these findings replicated successfully when applied to an alternate course environment. These findings suggest that spending more time in various course pages, posting more sophisticated forum posts or posting more frequently in the forums, and being willing to follow the pace set by the instructor made a learner more likely to complete the course regardless of course topic and design. Interestingly, two findings contradicted previously published results. A further study (Andres et al., 2018) extended this work and tested a subset of these findings (the findings not requiring survey data not available in most courses) across 29 sessions of 17 MOOCs, finding that 12 of the 15 findings relevant to MOOC completion replicated when applied to this larger dataset. By collecting this data and sharing it through the MORF platform with its privacy protections, MORF enabled large scale replication and validation of past scientific work without individual data agreements.

MORF was also used for replication study in (Gardner, Brooks, Andres, & Baker, 2018), which investigated findings from (Xing, Chen, Stein, & Marcinkowski, 2016). This work found that when applied to the broader range of MOOC data available in MORF, only some of Xing et al's findings replicated. In specific, findings from the initial paper that stacked ensemble classifiers performed better than single-algorithm classifiers failed to replicate in a majority of cases, and the paper found evidence suggesting that another analysis around appending week-by-week features only replicated in 2 of 9 comparisons. This work also articulated some challenges to replication of research where authors both fail to provide open materials and have published limited details of their work making it impossible to reproduce.

The framework's large data repository and open capabilities has also allowed for novel research. By leveraging the data repository researchers can also ask novel research questions and robustly test new hypotheses. One example of this is recent work studying how well prediction models of MOOC completion generalize across countries, in terms of the cultural and demographic variables (Andres-Bray, Hutt, & Baker, under review). This investigation examined model performance across learners from 81 countries and over 1.9 million learners. Models were developed to predict student dropout in each country in the dataset, and then were tested on every other country in the dataset. The researchers initially hypothesized that cultural factors would predict a model's degree of transfer between countries (using a popular model of national-level culture), and the results indeed suggested that a mis-match between the training and test country cultural features were associated with poorer model transfer. In specific, differences in power distance, individualism/collectivism, and long-term/short-term orientation were predictive.

5 Discussion and Conclusions

In this paper, we have presented ongoing work surrounding the development and implementation of the MOOC Replication Framework (MORF). MORF is a research architecture that supports scalable, replicable research, whilst respecting the privacy of learners. This is achieved through a platform that allows researchers to utilize large and comprehensive database of MOOC learners, including PII data, without ever directly accessing the data. MORF's approach achieves a different balance of risks and benefits than previous approaches to sharing data in education. Past approaches such as the PSLC DataShop, the ASSISTments data-sharing framework, and one-off data sharing in competitions made deidentified data sets available openly. This approach removes some key information that can be useful in analyses (such as demographic data and the ability to link students between data sets) while still presenting some reidentification risk. By offering restricted access on complete data, MORF enables a broader range of analyses and increases security, albeit at higher technical difficulty for users.

This work presents a blueprint for ethical data sharing in education. We recognize that PII can be critical in education research and in ensuring algorithmic fairness. Thus, researchers should have the ability to analyze their work with those fields included. Doing so must not compromise learner privacy. As noted in section 1.1, protecting student privacy can sometimes be a goal in conflict with goals of OpenScience and OpenData. Our approach attempts to satisfy all these objectives. MORF 2.1 supports OpenScience and replicability by providing researchers access to run complex analyses on learner data, but protects privacy by providing only indirect analysis. MORF 2.1 also has fewer limitations than previous iterations of MORF, resulting in a broader range of potential research questions that the framework can be used to investigate.

By leveraging cloud computing resources, this approach is able to build upon existing technology to create a protected database that facilitates complex data analysis. Though users are currently restricted in the scope of outputs that can be exported, we are actively working with users to adapt the framework to meet the needs of the userbase. By running all analyses in separate instances that cannot send out messages via the internet except through MORF's output functionality, we are able to protect student data and only provide access to the results of analysis rather than raw data files. By providing this kind of indirect access, we facilitate complex experiments that reflect the complexities of the data, without violating student privacy or risking reidentification.

Student privacy is further protected through the MORF API. All users must authenticate with a valid API key. Users must register in order to receive a key, and receive any and all results to the email address provided. This ensures that all Users are documented within MORF with valid contact details. The email service also keeps members of the MORF Administrators team informed of any errors so that they can provide support where appropriate.

This paradigm is only as powerful as the data included. We started this process with MOOCs, which lend themselves to this approach. For example, learning within MOOCs primarily occurs directly within the platform, and all learner interactions are logged in a database. By starting with MOOCs, we also leverage existing database design and documentation. This limits the additional contextual knowledge required for users already familiar with data from platforms such as edX and Coursera. Similarly, the data is self-contained. All activities are already recorded within the MOOC, and there is typically no need to merge multiple data sources (e.g., linking interaction data to teacher notes, paper tests, classroom observations, etc.). As such, importing additional years of data is relatively limited in the programmer time needed.

MOOCs do have some limitations in terms of their data relative to other types of learning activities, however. The range of actions available to learners is often limited to watching videos, answering short-answer activities, or posting in the forum. Specific MOOCs may include more complex assignments and activities (Aleven et al., 2018; O'Malley, Agger, & Anderson, 2015), but these features are not common across the full spectrum of MOOCs. This in turn limits the depth of discovery possible from the data available within this platform. There is also not typically an underlying student knowledge model that student performance is tagged with (compare to Aleven & Koedinger, 2013), making it harder to explicitly evaluate student learning over time or to replicate findings observed in other educational environments. The MORF paradigm's applicability is not limited to MOOCs, however. As noted above, MOOC data was a convenient first context to develop this infrastructure for. This approach could be expanded to include data from for-credit university courses, or even K-12 or professional training courses. The paradigm could even be extended further to consider data from intelligent tutoring systems, learning management systems, courseware simulations, and games. As we consider new approaches to ethical data sharing in education and supporting the goals of open science, we must continue to innovate on the notions exemplified in MORF. Our methodology allows researchers full access to data, while controlling the type of output they receive. We intend to continue to develop MORF to increase its capability and usefulness for the research questions of MORFs userbase.

5.1 Critical Challenges that Remain

5.1.1 Technical Challenges

A major challenge we are currently addressing is how to allow MORF users to access clickstream data efficiently. Clickstream data that is generated in MOOC platforms such as Coursera is usually in the tens of Gigabytes. Loading such data into memory and running operations such as parsing imply high time and space computational complexity. Depending on the data users seek to extract, it is sometimes practical to import clickstream data into a SQL database. However, some traditional operations are not suitable to run on clickstream data that has millions of rows. As is often the case when dealing with very large data sets, approaches such as MapReduce (Dean & Ghemawat, 2008) may be the most appropriate. We are currently exploring how to make use of this type of data processing models to make working with clickstream data more efficient for users and less costly for the platform.

A further technical challenge that we are working to resolve is in supporting users as they debug their code. Though we provide sample data for users to test their extraction and analysis programs, this sample data is not exhaustive in order to protect student privacy. As such, users may likely encounter errors when applying to the broader dataset resulting in a failed job. Currently only MORF technical team members get system-level error messages, and if the error requires code changes, the MORF team must work with the researcher. This approach does not scale to researchers debugging code individually. We are currently working to design a more extensive infrastructure for providing debugging detail without risking PII extraction.

Alongside these technical challenges, we intend to conduct more formal usability and security testing. This testing will take two forms. Firstly, formal usability testing, working with users to evaluate how easy it is to use MORF to conduct analyses, in order to make it easier to use effectively. In this work, we will take authentic research tasks and investigate whether it is feasible for participants to correctly and efficiently conduct those analyses. As part of this evaluation, we will compare users conducting analyses using MORF to conducting those same analyses with full data access (but still using Docker containers and other replicability best practices built into the environment). Such testing will provide a baseline for understanding the additional usability and time cost involved in using MORF, and we have already observed that typical exploratory data analysis is difficult to do if the user has no sample data. We anticipate that this testing will also involve user surveys and interviews to gain additional feedback on the process of using MORF. Secondly, we will conduct further security testing to ensure that users cannot extract any PII data, bringing in both security experts and regular users and asking them to find ways to extract PII when using MORF, with our regular security processes in place. This will go beyond the extensive security reviews we have already conducted with IT professionals; we will ask these users to actively attempt to defeat the platform's security. While we are confident in our framework, rigorous testing (and fixing any issues identified) will help further increase our confidence in the platform's security and reassure data partners.

5.1.2 Scientific and Ethical Challenges

A secondary area of critical challenge for enhancing MORF going forward is the need to support a broader span of the scientific process. In MORF version 2.1, any aggregate output is now possible (compared to limited functionality in 2.0), however, often researchers want to look at the content of specific data points to verify algorithm functioning or iterate an algorithm. For example, a researcher developing a linguistic algorithm (Crossley, Dascalu, McNamara, Baker, & Trausan-Matu, 2017) may want to look at specific text strings to see how they are classified. Or a researcher may want to look at the details of specific students being poorly classified by a dropout prediction algorithm (Gardner et al., 2018), in order to support a next pass on iterative feature engineering. In version 2.1, this is not possible.

Figuring out a better way to enable researchers to view some data while still maintaining full privacy will be a key challenge, not just for the MORF infrastructure, but for privacy-protecting research data infrastructures in general.

Eventually, it may be desirable to develop functionality for researchers without a background in data science. Currently, researchers submit jobs via the command line (although they can use IDEs and notebooks to create their job). Extending MORF through a graphical user interface like Tigris – or perhaps integrating an existing graphical data science platform such as RapidMiner, Orange, or Knime – could make MORF accessible to a much broader audience of researchers.

As we increase the scope of MORF and the possible research opportunities, there are increasing risks of introducing vulnerabilities to our paradigm. For instance, it may eventually become important to conduct analyses that involve linking multiple institutions' data together at the level of individual students. Creating data tunnels to enable this will create new challenges and risks to privacy that will demand careful work to ensure the maintenance of student privacy and institutions' legal obligations.

For the successful development of the MORF paradigm, we must be constantly evaluating the risks of data leak or data identification. In order to protect student privacy, there must be ongoing evaluation as functionality is added. Similarly, data encodings or structures may change as we add additional data to the repository, adding new risks. This again requires continual review and testing.

5.2 Future work

MORF is still an adapting platform, with development ongoing to enable more detailed research analysis. In addition to addressing some of the challenges outlined above, we will continue to add additional data to the MORF repository from existing partnerships, and hope to forge new partnerships moving forward, both with institutions and researchers. We hope that this expansion can also include data collected by a diverse set of institutions who offer learning in unique contexts. Currently, all the data in the primary installation of MORF 2.1 is from the University of Pennsylvania (with the University of Michigan planning an upgrade from 2.0 to 2.1), and all of the content is taught in english. This U.S./English language focus may limit potential applications and research questions – even though our learners are worldwide, MOOC platforms and courses differ in design and content across countries. As MORF grows and the opportunity for new installations arises, it is our hope that this paradigm will be adopted at other universities. We welcome new collaborations, and work is ongoing to provide additional support and resources for research partners looking to use MORF and to provide increased functionality for researchers whilst preserving student and instructor privacy.

Another avenue of future work is to apply our paradigm to other data sources, such as educational games, and online learning environments for K-12 and undergraduates. Multiple areas of education scholarship could benefit from increased access to data, but any such access must protect student privacy as MORF does. It is our hope that this framework may be applied to new domains within education. Though this will present challenges of how to unify data across platforms, it will provide increased opportunity and accessibility to education research.

REFERENCES

Aleven, V., & Koedinger, K. R. (2013). Knowledge component (KC) approaches to learner modeling. *Design Recommendations for Intelligent Tutoring Systems*, *1*, 165–182.

Aleven, V., Sewall, J., Andres, J. M., Sottilare, R., Long, R., & Baker, R. (2018). Towards adapting to learners at scale: integrating MOOC and intelligent tutoring frameworks. *Proceedings of the Fifth Annual ACM Conference on Learning at Scale*, 1–4.

Andres-Bray, J. M., Hutt, S., & Baker, R. S. (Under Review). *Exploring Cross-Country Prediction Model*

Generalizability in MOOCs.

Andres, J. M. L., Baker, R. S., Gašević, D., Siemens, G., Crossley, S. A., & Joksimović, S. (2018). Studying MOOC completion at scale using the MOOC replication framework. *Proceedings of the 8th International Conference on Learning Analytics and Knowledge*, 71–78.

Andres, J. M. L., Baker, R. S., Siemens, G., Gašević, D., & Spann, C. A. (2017). Replicating 21 findings on student success in online learning. *Technology, Instruction, Cognition, and Learning*, *10*(4), 313–333.

Arroyo, I., Burleson, W., Tai, M., Muldner, K., & Woolf, B. P. (2013). Gender differences in the use and benefit of advanced learning technologies for mathematics. *Journal of Educational Psychology*, *105*(4), 957.

Baker, R. S., & Hawn, A. (2021). Algorithmic Bias in Education. *International Journal of Artificial Intelligence in Education*, 1–41.

Baker, R. S., & Inventado, P. S. (2014). Educational data mining and learning analytics. In *Learning analytics* (pp. 61–75). Springer.

Boettiger, C. (2015). An introduction to Docker for reproducible research. *ACM SIGOPS Operating Systems Review*, *49*(1), 71–79.

Chakraborty, S., Tomsett, R., Raghavendra, R., Harborne, D., Alzantot, M., Cerutti, F., … Gurram, P. (2018). Interpretability of deep learning models: A survey of results. *2017 IEEE SmartWorld Ubiquitous Intelligence and Computing, Advanced and Trusted Computed, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People and Smart City Innovation*. https://doi.org/10.1109/UIC-ATC.2017.8397411

Chatzimparmpas, A., Martins, R. M., Jusufi, I., & Kerren, A. (2020). A survey of surveys on the use of visualization for interpreting machine learning models. *Information Visualization*, *19*(3), 207–233.

Consortium, S. D. P., & others. (2018). *Student Data Privacy Consortium: Policy and procedures*. SDPC. Retrieved from: https://privacy. a4l. org/wp-content/uploads/2018/06~….

Corrin, L., Kennedy, G., French, S., Buckingham Shum, S., Kitto, K., Pardo, A., … Colvin, C. (2019). The ethics of learning analytics in Australian higher education. *Accessed Online*, *26*.

Crossley, S., Dascalu, M., McNamara, D. S., Baker, R., & Trausan-Matu, S. (2017). Predicting success in massive open online courses (MOOCs) using cohesion network analysis. *Proceedings of the International Conference on Computer-Supported Collaborative Learning*, 103–110. Philadelphia, PA: International Society of the Learning Sciences.

Dean, J., & Ghemawat, S. (2008). MapReduce: simplified data processing on large clusters. *Communications of the ACM*, *51*(1), 107–113.

Echtler, F., & Häußler, M. (2018). Open source, open science, and the replication crisis in HCI. *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*, 1–8.

EdX. (2020). Using the Research Data Exchange Data Package — EdX Research Guide documentation. Retrieved November 28, 2021, from https://edx.readthedocs.io/projects/devdata/en/latest/rdx/index.html

EdX. (2020). EdX Researgh Guide. Retrieved from https://edx.readthedocs.io/projects/devdata/en/latest/using/package.html

Fecher, B., & Friesike, S. (2014). Open science: one term, five schools of thought. *Opening Science*, 17–47.

Finkelstein, S., Yarzebinski, E., Vaughn, C., Ogan, A., & Cassell, J. (2013). The effects of culturally congruent educational technologies on student achievement. *International Conference on Artificial Intelligence in Education*, 493–502.

Foster, E. D., & Deardorff, A. (2017). Open science framework (OSF). *Journal of the Medical Library Association: JMLA*, *105*(2), 203.

Frankowski, D., Cosley, D., Sen, S., Terveen, L., & Riedl, J. (2006). You are what you say: privacy risks of public mentions. *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 565–572.

Fyfe, E. R., de Leeuw, J. R., Carvalho, P. F., Goldstone, R. L., Sherman, J., Admiraal, D., … Motz, B. A.

(2021). ManyClasses 1: Assessing the Generalizable Effect of Immediate Feedback Versus Delayed Feedback Across Many College Classes. *Advances in Methods and Practices in Psychological Science*, *4*(3), 25152459211027576. https://doi.org/10.1177/25152459211027575

Gardner, J., & Brooks, C. (2018). Dropout model evaluation in MOOCs. *Thirty-Second AAAI Conference on Artificial Intelligence*.

Gardner, J., Brooks, C., Andres, J. M., & Baker, R. (2018). Replicating MOOC predictive models at scale. *Proceedings of the Fifth Annual ACM Conference on Learning at Scale*, 1–10.

Han, F., & others. (2014). *Modeling Problem Solving in Massive Open Online Courses*. Massachusetts Institute of Technology.

Han, F., Veeramachaneni, K., & O'Reilly, U.-M. (2013). Analyzing millions of submissions to help MOOC instructors understand problem solving. *NIPS Workshop on Data Driven Education*, 1–5.

Heffernan, N. T., & Heffernan, C. L. (2014). The ASSISTments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. *International Journal of Artificial Intelligence in Education*, *24*, 470–497. https://doi.org/10.1007/s40593-014-0024-x

Howard, T., & Terry Sr, C. L. (2011). Culturally responsive pedagogy for African American students: Promising programs and practices for enhanced academic performance. *Teaching Education*, *22*(4), 345–362.

Kizilcec, R. F., & Lee, H. (2020). Algorithmic fairness in education. *ArXiv Preprint ArXiv:2007.05443*.

Kizilcec, R. F., Pérez-Sanagustin, M., & Maldonado, J. J. (2017). Self-regulated learning strategies predict learner behavior and goal attainment in Massive Open Online Courses. *Computers & Education*, *104*, 18–33.

Koedinger, K. R., Baker, R. Sj., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the EDM community: The PSLC DataShop. *Handbook of Educational Data Mining*, *43*, 43–56.

Liu, R., Koedinger, K., Stamper, J., & Pavlik, P. (2017). Sharing and Reusing Data and Analytic Methods with LearnSphere. *Workshop and Tutorials Chairs*, 475.

O'Malley, P. J., Agger, J. R., & Anderson, M. W. (2015). Teaching a chemistry MOOC with a virtual laboratory: Lessons learned from an introductory physical chemistry course. *Journal of Chemical Education*, *92*(10), 1661–1666.

Ostrow, K. S., & Heffernan, N. T. (2016). Studying learning at scale with the ASSISTments TestBed. *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, 333–334.

Paquette, L., Baker, R. S., & Moskal, M. (2018). A system-general model for the detection of gaming the system behavior in CTAT and LearnSphere. *International Conference on Artificial Intelligence in Education*, 257–260.

Pardos, Z. A., & Heffernan, N. T. (2010). Modeling individualization in a bayesian networks implementation of knowledge tracing. *International Conference on User Modeling, Adaptation, and Personalization*, 255–266.

Pardos, Z. A., & Kao, K. (2015). moocRP: An open-source analytics platform. *Proceedings of the Second (2015) ACM Conference on Learning@ Scale*, 103–110.

Patikorn, T., Heffernan, N. T., & Baker, R. S. (2018). ASSISTments Longitudinal Data Mining Competition 2017: A Preface. *Proceedings of the Workshop on Scientific Findings from the ASSISTments Longitudinal Data Competition, International Conference on Educational Data Mining*.

Politou, E., Alepis, E., & Patsakis, C. (2018). Forgetting personal data and revoking consent under the GDPR: Challenges and proposed solutions. *Journal of Cybersecurity*, *4*(1), tyy001.

Pournaras, E. (2017). Cross-disciplinary higher education of data science--beyond the computer science student. *Data Science*, *1*(1–2), 101–117.

Prinsloo, P., & Slade, S. (2017). An elephant in the learning analytics room: The obligation to act. *Proceedings of the Seventh International Learning Analytics \& Knowledge Conference*, 46–55.

Ro, H. K., & Loya, K. I. (2015). The effect of gender and race intersectionality on student learning

outcomes in engineering. *The Review of Higher Education*, *38*(3), 359–396.
Selent, D., Patikorn, T., & Heffernan, N. (2016). Assistments dataset from multiple randomized controlled experiments. *Proceedings of the Third (2016) ACM Conference on Learning@ Scale*, 181–184.
Stamper, J., Koedinger, K., Pavlik Jr, P. I., Rose, C., Liu, R., Eagle, M., & Veeramachaneni, K. (2016). Educational data analysis using LearnSphere workshop. *Proceedings of the EDM 2016 Workshops and Tutorials Co-Located with the 9th International Conference on Educational Data Mining. Raleigh*.
Sun, D., Mao, Y., Du, J., Xu, P., Zheng, Q., & Sun, H. (2019). Deep learning for dropout prediction in MOOCs. *2019 Eighth International Conference on Educational Innovation through Technology (EITT)*, 87–90.
U.S. Department of Labor. (n.d.). Guidance on the Protection of Personal Identifiable Information | U.S. Department of Labor. Retrieved November 28, 2021, from https://www.dol.gov/general/ppii
van der Zee, T., & Reich, J. (2018). Open education science. *AERA Open*, *4*(3), 2332858418787466.
Wicherts, J. M., Borsboom, D., Kats, J., & Molenaar, D. (2006). The poor availability of psychological research data for reanalysis. *American Psychologist*, Vol. 61, pp. 726–728. Wicherts, American Psychological Association. https://doi.org/10.1037/0003-066X.61.7.726
Wolins, L. (1962). *Responsibility for raw data*.
Xing, W., Chen, X., Stein, J., & Marcinkowski, M. (2016). Temporal predication of dropouts in MOOCs: Reaching the low hanging fruit through stacking generalization. *Computers in Human Behavior*, *58*, 119–129.
Yacobson, E., Fuhrman, O., Hershkovitz, S., & Alexandron, G. (2021). De-identification is Insufficient to Protect Student Privacy, or--What Can a Field Trip Reveal? *Journal of Learning Analytics*, *8*(2), 83–92.

Biographies

Stephen Hutt is a postdoctoral researcher at the University of Pennsylvania and Assistant Director of the Penn Center for Learning Analytics. His research interests lie at the intersection of Artificial Intelligence, Learning Sciences, and Cognitive Science, specifically, considering how we can leverage AI to improve education and our understanding of learning.

Ryan S. Baker is an Associate Professor in the Graduate School of Education at the University of Pennsylvania, the Director of the Penn Center for Learning Analytics, the founding President of the International Educational Data Mining Society, the Associate Editor of the Journal of Educational Data Mining, and the Editor of Computer-Based Learning in Context.

Michael Mogessie is a project scientist at Carnegie Mellon University who has several years of experience in developing educational and other software in both academia and industry. He has more recently led educational technology projects to implement cognitive tutors in games and other educational software.

Miggy Andres-Bray is a Senior Data Scientist at McGraw Hill Education. His research interests are in Learning Analytics, MOOCs, and Replication. He recently completed his Ph.D. in Education, where he investigated the cross-cultural replicability of completion prediction models across millions of learners from over 80 different countries using the MOOC Replication Framework (MORF).

Christopher Brooks is an Assistant Professor at the University of Michigan who builds and studies the effects of educational technologies in higher education and informal learning environments with a particular domain focus on data science education and methodological interests in predictive modeling, learning analytics, and collaborative learning.

STRUCTURED PRACTITIONER NOTES

What is already known about this topic

- Personal Identifying Information (PII) has many valid and important research uses in education
- The ability to replicate or build on analyses is important to modern educational research, and is usually enabled through sharing data
- Data sharing generally does not involve PII in order to protect student privacy
- MOOCs present a rich data source for education researchers to better understand online learning

What this paper adds

- The MOOC replication framework (MORF) 2.1 is a new infrastructure that enables researchers to conduct analyses on student data without having direct access to the data, thus protecting student privacy
- Detail of the MORF 2.1 structure and workflow

Implications for practice and/or policy

- MORF 2.1 is available for use by practitioners and research with policy implications
- The infrastructure and approach in MORF could be applied to other types of educational data

Acknowledgements

This research was supported by the National Science Foundation (NSF) (NSF-OAC#1931419) and the Penn Center for Learning Analytics, at the University of Pennsylvania. Any opinions, findings and conclusions, or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the NSF. We thank the Online Learning Initiative for their support with this project.

Statement on Open Data, Ethics, and Conflict of Interest

The data available in the MOOC Replication Framework can be used by external researchers through the framework under a data use agreement with the relevant university which oversees the data being accessed, and with evidence of approval of the research by an Institutional Review Board (or comparable ethical oversight organization, such as in uses of MORF for non-US data by non-US researchers). All human subjects research discussed in this paper was conducted under the oversight of an Institutional Review Board. No conflict of interest (financial or non-financial) is declared by the authors.