Student 1

3: (strong accept)
Paper [13]: Enabling real-Time adaptivity in MOOCs with a personalized next-step recommendation framework

Major contributions

The quest for developing more real-time adaptive MOOCs has been hampered by the lack of enough real-time learner data across platforms and the lack of sophisticated recommendation models that accommodate the learning context as well as the complexity of the subject. In this paper, the authors address these two challenges and make three contributions to adaptive personalization to MOOCS namely:

1) They provide a JavaScript solution to solve the problem of logging real-time learner events needed for data-driven interventions
2) They introduced a novel LSTM based behavioral model that can predict the next page a learner will most likely spend a significant time on - their model outperforms the existing baselines.
3) They provide a proof of concept by incorporating the data-driven recommendation framework in a live MOOC

Main strengths

- One of the major strengths of the paper is that it provides a solution for real-time interventions in MOOCs which is very timely and relevant for the online education field.
- Secondly, all technical solutions proposed by the authors to deploy these interventions can be achieved without needing a modification to open-edx; rather they only require access to the instructor or the design team to edit course material.
- Their explanations are very thorough regarding platform augmentation, modeling as well as real-time deployment especially when it comes to the choices they made and why which is very useful for the relevant researchers and also novice readers.
- They compare their model with several baselines.

Main weaknesses

- Since the authors use the raw time series data (as they don't want the model to make any assumptions about the behavior) to feed the model, that raises the issue of interpretability of the behaviors as to “which behaviors are better at predicting the next page a user was likely to spend significant time on”.
- It is uncertain how they define what should be the “significant amount of time” i.e. desirable time bucket range based on which they decide either to give the recommendation or not?
- There is no discussion on the performance results of the models. For ex, why do the authors think the baseline LSTM performs similarly to the proposed 4 other combinations of LSTM with time incorporated?

- The authors did not provide the motivation of choosing time-bucketed-input with time-concatenated-output model when the time-bucketed-input with non-concatenated output seems to perform best.
- It would be interesting to see a formal evaluation of the “recommendation system” by the users of the system, i.e. was the system indeed recommending what the user wanted? A simple calculation of how many times the user “accepted” the recommendation?

Connection to the bigger picture

Millions of people around the world rely on MOOCs to learn various subjects. However, a majority of them do not end up finishing the course due to feeling lost or left without enough guidance. The authors address a very relevant problem in that regard by providing a sense of direction/recommendations to the user in real time. As the proposed model is based on what a learner may want (or end up spending a significant time on); It would be very interesting to see if what learners want is what they also need to achieve the educational goals of the course to which the authors point out as a part of their future work.

Student 2

2: (accept)
Summary
In this paper, the authors attempt to predict what page will the student want to spend a significant amount of time on. The incentive behind predicting the URL students will most likely visit is to optimise their use of MOOCS on platforms such as Coursera and EdEx. Indeed, navigating a MOOC can be quite daunting, and students may lose engagement without any guidance.
Because it has been shown that students react best to such recommendations when they are bored, those recommendations can come at the optimal time rather than at a constant rate. In this paper, they propose a novel way to integrate those recommendations into the EdEx platform, they develop a recommendation system based on an LSTM, and construct a prototype from those two novel approaches.

Strengths
- It was clear from the paper that they put a lot of efforts in order to understand the data in depths. The data processing and cleaning showed a deep reflection about what outliers and the values in general could mean from a user point of view.
- In order to make their recommendation effective, they not only took into account the technical aspect of page recommendation, but also the HCI component that proposed those recommendations to the user
- They propose an approach to help students stay engaged in a MOOC, which I think is a big issues in most online courses (to stay engaged and finish the online course), the incentive behind developing such a method is therefore very much appreciated, especially in view of current events.
- There was a lot of time spent on how they developed and implemented the recommendations on the platform, which showed how teahers and course designers could actually user their research without having to fiddle with the platform for hours on end. Additionally, it also helps understanding their logic behind their recommendations, and it was in accordance with what they developed in related work.
- Implementing their recommendation algorithms via an LSTM was really clever

Weaknesses
- I was confused by some of their decisions to take out data points such as quizzes. Aren't students who pass their quizzes, or who have passed different quizzes, interested in different pages than those who actually did ?
- I don't understand why they came up with a baseline that does not include time input if their goal is to predict the pages students are the most likely going to spend a lot of time on (they need to make a prediction about time too).
- I have a hard time understanding how they measure accuracy. The output is not per se binary, and it is very subjective, as even though a student did not go on the recommended page, it could still be a relevant page. Additionally, their could be multiple optimal page at certain point in time. Although the software and model implementations were described in detail, I found that the way they computed their performances was a bit obscure and under detailed compared to the rest of the paper

Connection to the bigger pictures.
Because of the current situations, MOOCs, or online courses are becoming the norm. Some of those offered courses are sometimes hard to navigate, or full of recommended, advised, optional readings, and it is hard to know what is relevant for us at any point in the course, or hard to find what we should be looking for. Implementing such a module into our own courses and platforms (moodle, switchtube, ...) could have a high impact, and could maybe retain our attention better.
In terms of classis moocs, it is nice to find techniques enhancing the engagement of the students, and facilitating the navigation of the students through the MOOC, helping them learning better and in a more optimal manner. I could really see how such an improvement made on diverse moocs and online courses could benefit a lot of students.

Student 3

3: (strong accept)
Main contributions
-Creating a real-time event logging to record navigational events
-Creating a recommendation system that predicts which page the user is going to spend time on without being a time-out event.
-Testing their model and implementation on a live MOOC

Major strengths
-The edX platform was augmented to track real-time behavior and adapt content without altering the open-edX codebase.
-Implementing the JavaScript logger. The description is very clear, it is nice to have code snippets and a database entry sample.
-Hiding the container with the recommendation so that the user does not notice when something goes wrong in the pipeline.
-Clear and nice explanation of the choice of technology and course. Very easy to replicate and to understand the architecture.
-Complete details about the hyperparameters tested and the variations in the architecture.
-Meaningful baselines including the n-gram
-Taking into account changes in the course by adding a URL to control for the version and future changes.

Major weaknesses
-The time between pages was approximated by taking the time difference before accessing the next URL. What if someone left the page open and got a coffee (time-out event)? How do the distributions look? What was the treatment of the outliers? I'm guessing the outliers were the ones that spent more than 1,800 seconds (30 minutes) on the page and in one model they are put into the biggest bucket and in the normalized-time-input model, they are taken as 1 and then the model predicts for lengths less than 30 minutes but it would be useful to know how many outliers were there and how the time spent is distributed. Why 30 minutes and not 20 or 25? How would the model behave differently?
-It is interesting that the system does not differentiate between successful (certificate earners) navigation strategies and the strategies of the students that did not complete the course. Is it possible that by not adding the bias, the system is recommending the non-optimal patterns? What if the system is suggesting content that will make the student feel lost just like past students did?
-If the purpose is not to get lost, then wouldn't it be more interesting to recommend "previous step" or having links to review the previous content? Seems like the "next-steps" could be used to raise engagement levels in advanced students but not help students likely to fall behind.
-Privacy. It is possible to retrieve the non-anonymized userID.
-The Web service was created with Flask which does not scale and should not be used in production.
-Are 9,172 learners enough? The median number of sequences was 6, seems very small for a LSTM.

Connection to the bigger picture
-MOOCs can provide high-quality education at little or no cost to many internet users. However, completion rates are very low. Students report being non-engaged and lost. The importance of this

paper in the big picture is that integrating a recommendation system into MOOCs can provide a personalized experience for students and hopefully raise engagement levels and provide the content that the students will be the most interested in. For future research, it would be interesting to adjust the difficulty to each student and recommend previous as well as the next steps so that students can go back and revise different topics.

Student 4

3: (strong accept)
CS-702: Week 6

Review of Pardos et al. 2017

Enabling Real-Time Adaptivity in MOOCs with a Personalized Next-Step Recommendation Framework

Summary

In this paper, Pardos et al. present a solution to accompany MOOC learners. They do this by addressing two obstacles confronting learners in MOOCs: (i) insufficient support of real-time learner data across platforms and (ii) lack of maturity of recommendation models. In order to do this, they augment the edX platform with a real-time logging and dynamic content presentation solution, and develop a recommendation model using RNNs, implementing it on five different LSTM models. They train their models with real data and select the best-performing model to be piloted in a live course.

Strengths

This paper has several strengths, the most important being the fact that it is the first example of delivering a data-driven personalised learning experience in a MOOC. Not only is it the *first*, but I also see that the paper was well thought out and validated on a well-known platform (edX). The fact that the authors gave an on-point, succinct review of the literature, and justified most of the design decisions they took, strongly supported their approach to engineering, tuning, and testing their recommender system. Regarding their implementation, I think that it is very important that it does not require modifying the edX source code, which would introduce a high barrier to entry for their solution. Furthermore, they provide the full client-side code, which provides transparency and reproducibility. It is also worth highlighting that they base their strategy on the strong relationship between the success of MOOC learners and the learning path they took through the course. This provides a strong justification of the need for a solution like theirs. Furthermore, the fact that they do not assume that there is a correct or incorrect learning path is a strength, as it makes their system applicable to a wider variety of courses and learning styles. Finally, the fact that they trained their model on five LSTM networks and three baselines provides a more robust grounding for their results. This approach to training and testing, as well as their pilot validation with a live course, whose selection was fully justified using the entropy metric, make this paper particularly solid.

Weaknesses

Although this paper is strong overall, there are a few weaknesses that stand out in the text. First, the code that they propose is susceptible to both changes in the edX platform and changes in the course structure. Although the authors acknowledge this, I believe that it poses a big challenge for implementation by other parties. Hence, although the work can pave the way for this type of recommender systems for MOOCs, the actual system they propose needs further improvement to be production-ready. Second, I believe their system architecture is a bit more complex than necessary. Without going into detail, perhaps the Python web service could be merged with the API.

The idea is that complex systems will most likely be hard to implement by third parties. That being said, if they would provide ready-made orchestrated versions of their software (e.g. using Docker and Kubernetes), this problem might be solved. I also have minor issues with the sample code bits being messy (inconsistent variable declarations and indents), as it does not give a good first impression of how the software was engineered. Nevertheless, they are appreciated and helped me understand what they were doing without looking at the full codebase.

Connection to the Bigger Picture

The authors note in their paper that MOOCs suffer from dismal completion rates. This means that their solution has an application to real use cases, where it would have a direct effect on supporting both students and MOOC platform providers. Given my personal experience with MOOCs and how hard it is to engage with some of the content over a long period of time, I appreciate the value that this work has and understand its importance.