Midterm Report
Intelligent Tutoring System: Outer Loop

Harsimran Bath (COMS W3995 sec 028) and Weiman Sun (COMS E6901 sec 028)

February 2, 2018

1 Abstract

In our project, we have completed the following user stories: Outer Loop Design - Teachers (#126), Outer Loop Design - Students (#126). In addition, we are actively working on Outer Loop Feature Distillation (#127) and Outer Loop Implementation - Students (#129). Data issues initially set us back (more information in Limitations & Assumptions), but we have now worked around them and are now making good progress on the implementation. In this report, you will find information on Architecture, Implementation, Limitations & Assumptions, and Future work.

2 Architecture

Our architecture is consistent with the our proposal for the most part. We plan on having an adapter, which will act as an interface around the recommendation engine and potentially where preprocessing and feature distillation can be done, as well as, pulling mLab data.

There are, however, two things worth pointing out:

1. The recommendation engine is being developed in Python because of good learning and mathematical algorithms support. We will use AWS Lambda Functions to expose our Python recommendation methods as API endpoints, which then the Adapter can interact with. (More information in implementation.)

2. The data models are currently hard-coded and data is manually generated. Once good, these models can be merged into Mlab. (More information in the Limitations & Assumptions section.)

3 Implementation

We have begin implementing our code in Python, as Python offers significant APIs support for learning algorithms and mathematical operations.

So far, we have implemented the first part of recommendation algorithm. The first part of the algorithm is to find the similar users for a given user. In the second part of the algorithm, we recommend courses from the courses that the similar users have taken but the given user has not taken. When finding similar users, we employ the similarity computation in the collaborative filtering method. The formula is:

$$sim(a, b) = \frac{\sum_{p \in P}(s_{a,p} - \bar{s}_a)(s_{b,p} - \bar{s}_b)}{\sqrt{\sum_{p \in P}(s_{a,p} - \bar{s}_a)^2}\sqrt{\sum_{p \in P}(s_{b,p} - \bar{s}_b)^2}} \tag{1}$$

where a, b are two users, $s_{b,p}$ is the score of user b in course p, and P is the set of courses taken both by a and b. Each course has three features: difficulty, average ctScores, and duration, all of which ranges from 0 to 3. The score in our case is the average of the above three features. In summary, in the first part of recommendation algorithm, we use formula (1) to compute the similarity between two users, and then rank the similarity and find the most k (user-defined) similar users.

4 Limitations & Assumptions

We have encountered a lot of issues around existing data models and inconsistent data within mLab. We spent first couple weeks of our project trying to understand the data and extract the features that would best help us. We had trouble understanding the structure and pinpointing features we could use. Gradually, through Jeff's and Johan's help, we began understanding everything more.

We have since come to the conclusion that it would be best to create our own data models for our project, so we can focus on our project and make progress. Once completed, our models can be merged back into mLab. This way our models are mature before they're added into mLab, and helps in maintaining mLab as well.

5 Future Work

We hope to finish Outer Loop Implementation - Students (#129) soon and begin working on Outer Loop Implementation - Teachers(#128). Simultaneously, we are addressing Outer Loop Feature Distillation (#127) through the adapter, and eventually, Outer Loop Feature API (#345). We expect the Feature distillation to not require much work, other than, perhaps, feature scaling. The API, we also imagine, will be fairly straightforward, not too time consuming. After all these, we will look into the Evaluation (#130), which we might not be able to complete this semester due to its complexity and magnitude.