Capstone Project Proposal
Aaron Moss

1. Domain Background

Riiid AIEd Challenge 2020
Source: https://www.kaggle.com/c/riiid-test-answer-prediction/overview

“In 2018, 260 million children weren't attending school. At the same time, more than half of these young students didn't meet minimum reading and math standards. Education was already in a tough place when COVID-19 forced most countries to temporarily close schools. This further delayed learning opportunities and intellectual development. The equity gaps in every country could grow wider. We need to re-think the current education system in terms of attendance, engagement, and individualized attention.

Riiid Labs, an AI solutions provider delivering creative disruption to the education market, empowers global education players to rethink traditional ways of learning leveraging AI. With a strong belief in equal opportunity in education, Riiid launched an AI tutor based on deep-learning algorithms in 2017 that attracted more than one million South Korean students. This year, the company released EdNet, the world’s largest open database for AI education containing more than 100 million student interactions.”

I have chosen this competition due to its close nature to my current position at an EdTech company. We provide content, assessments, and facilitate the interactions between educator & nursing student within their institution via SaaS.

2. Problem Statement

The competition is to develop a model that can predict the likelihood of a student getting an item correct. This competition will have new users outside of the training set but not new question items.

3. Datasets and Inputs:

Data sets can be viewed: https://www.kaggle.com/c/riiid-test-answer-prediction/data

Train.csv: Roughly 365,000 users in training set and over 100 million interactions between questions, explanations, and lectures.
Test.csv: Same columns as train.csv
Questions.csv: Over 13,000 questions with type of questions, grouped question ids, & content/topic tags
Lectures.csv: Over 100 lectures with tag/topics associated, type of lecture delivery, and connected to concept or solving question

There are 10 features in the training set with the discrete target (‘answered_correctly’) being ‘1’ for correct and ‘0’ for incorrect. Note: ‘-1’ is for lectures that the user consumed for that event. Simple EDA provided that the classes are well-balanced (~50% each class).

4. Solution Statement

The proposed solution to this problem is to apply tree-boosting classification model within Kaggle notebook or RNN time series within AWS Sagemaker instance. For train/test split, I will be simulating the batch inference by splitting the training set by last 100 interactions per user and, within the 100 observations, the last four data points will become the test set. In addition, I will need to build robust code to transform incoming batch to update user/item dataframes and merge back to original batch rows for inference. The expected output should be a range between 0.00 and 1.00 as it is a probability.

5. Benchmark Model

I will be using a native model as benchmark, in which, each user has a fixed probability of producing the correct answer using the training set's overall average across all users.

6. Evaluation Metrics

The evaluation metric for this competition is be Area under the Curve, which indicates the model's ability to correctly identify a random observation to the actual Class group (0= Miss, 1= Correct). However, I will be optimizing the model and features for binary LogLoss metric to help enhance model's target metric.

7. Project Design

Reduce training set memory allocation

The first step will be to get the appropriate question interactions from users in the train.csv. Secondly, due to its overall size, we will need to reassign dtypes to lower memory demand in train.csv.

Transform and feature build on supplement datasets

We need to transform questions.csv and lecture.csv which to be to dummy specific variables or label encode topic sequences for items.

EDA and Feature Engineering

Merge new supplement dataframes to training set**.** Once complete training set is produced, I will use descriptive statistics and EDA to find possible features. One possible feature will be mean encoding for content_id (question) and/or each user.

Data Splitting

Since this is a time-series problem, I will pull the last 100 user:qts pair interactions. I will use the last four of the 100 interactions as the test set to simulate the evaluation task.

Model training and evaluation

I will build the simple model and test out the evaluation metrics as baseline. Then I will try the improved features and different architectures w/ hyperparameters to reach improved evaluation metric.

Time Series API for Test set

I will predict which questions each student is able to answer correctly. Build a robust API call to receive, transform, and infer predictions. Debug issues once the Test API is running through my inference loop. The competition describes the Test API as it will perform batch inference and requires a prediction before going to the next batch.