CS 229 Project Report: Modeling Student Learning in Mobile App with Machine Learning

Zhaolei (Henry) Shi

June 12, 2019

Abstract

This study applies machine learning to student-question interaction data from a mobile app to model students' learning trajectories. Since student learning is unobserved, machine learning techniques are used to take student-question interactions as inputs and predict achievement (operationalized as expected probability of correct answer across all questions). The baseline methods include time-weighted naive Bayes and a question difficulty-modulated version of the same algorithm. The main model is a Siamese-like neural network that can capture students' questions-level performance across time. The best performing Siamese model was able to produce an accuracy rate of 87.3% with an F1 score of 0.922. Compared to the baseline models, the Siamese-like neural network also produces more consistent predictions of achievement ranking between students over time.

1 Motivation

Online education services promise to revolutionize education through unparalleled access and efficiency. While high profile MOOCs platforms have garnered press coverage, many experience difficulty in finding a profitable business model (McPherson and Bacow, 2015). In China, thanks to its large homogeneous market for education services, education technology companies have made headway in creating profitable business models.

This study applies machine learning to a new data source coming out of China. Specifically, I attempt to model students' learning trajectories using student-questions interactions in a mobile app designed to replace traditional paper-based homework assignments with in-app questions.

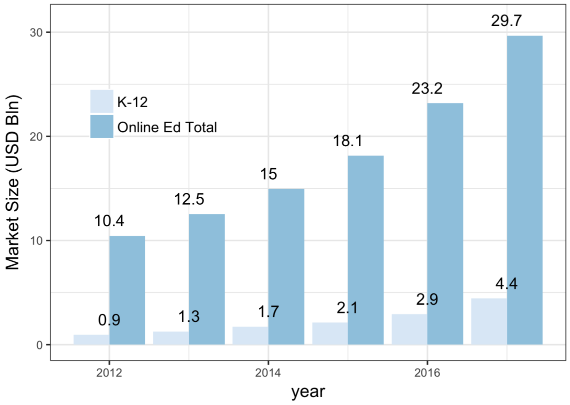


Figure 1: The K-12 Afterschool Education Market in China (**?**)

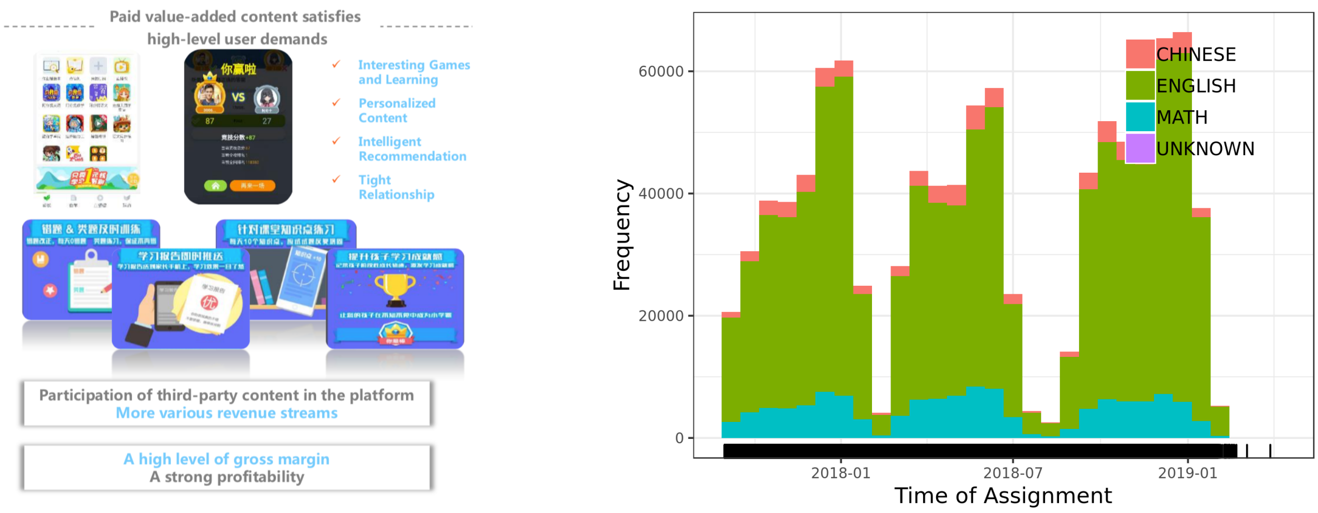


Figure 2: App Features and Homework Assignments Over Time

In the context of my study, the space of possible questions that a student can be exposed to is large. School teachers pick from a large question bank a set of questions for each assignment. Figure 1 shows the number of homework assignment issues on the platform over time. Critically, **each student is only exposed to a small subset of problems** in any given homework session. Hence, we do not observed student's performance on homework questions she did not see. As a result, we would be unable to compare one student's learning against of that of another. Therefore, the gaol of this study is to leverage machine learning techniques to **predict the students' performance on the questions that she did not see**.

2 Methods

2.1 Baseline Models

There is a large space of homework questions that a student can be exposed to as the company provides a large question bank. Each student is only exposed to a small subset of problems in any given homework session. The goal is to produce a measure of student achievement that reflects the students ability to answer correctly questions in expectation. I leverage machine learning methods way to extract this target measure from high dimensional sparse inputs.

A baseline model for student learning is one based on Naive Bayes. I assume that the probability of student i answering question j correctly is the same across all questions $\Pr(y_{i1} = 1) = \Pr(y_{i2} = 1) = \ldots = \Pr(y_{im} = 1)$. Hence, the Naive Bayes approach is to replace $\Pr(y_i = 1)$ with its empirical distributional equivalent which is

$$\hat{\Pr}(y_i = 1) = \frac{\sum_j \mathbb{1}\{y_{ij} = 1\}}{\sum_j \mathbb{1}\{i \text{ attempted } j\}}$$

To account for student achievement changes across time ($\Pr(y_i = 1)$ may be a function of time t), I weight the observations by a function of the distance in time between observation and prediction:

$$\hat{\Pr}(y_i = 1; t) = \frac{\sum_j \gamma(t_{ij} - t) \cdot \mathbb{1}\{y_{ij} = 1\}}{\sum_j \gamma(t_{ij} - t) \cdot \mathbb{1}\{i \text{ attempted } j\}}$$

where $\gamma(.)$ is a kernel function (e.g. Gaussian kernel).

A quick extension to this model is to have each question difficulty captured by a parameter θ_j. I call this the difficulty-modulated naive Bayes model. In this model, I assume $\Pr(y_{ij} = 1) = \theta_j \Pr(y_i = 1)$ where $0 \leq \theta_j \leq 1$. In this case, the naive Bayes estimates need to satisfy the following equations

$$\hat{\theta}_j = \frac{\sum_i \mathbb{1}\{y_{ij} = 1\}}{\sum_i \mathbb{1}\{i \text{ attempted } j\}}$$

$$\hat{\Pr}(y_i = 1; t) = \frac{\sum_j \gamma(t_{ij} - t)\mathbb{1}\{y_{ij} = 1\}/\hat{\theta}_j}{\sum_j \gamma(t_{ij} - t)\mathbb{1}\{i \text{ attempted } j\}}$$

For questions that were not answered correctly by any students, I use Laplace smoothing which is equivalent to adding one observation of a correct response to all the questions when computing their difficulty. All else equal, I expect that taking difficulty into account in the naive Bayes model will produce less noisy measurements of student achievement than the unmodulated version.

2.2 Siamese-Like Neural Network Model

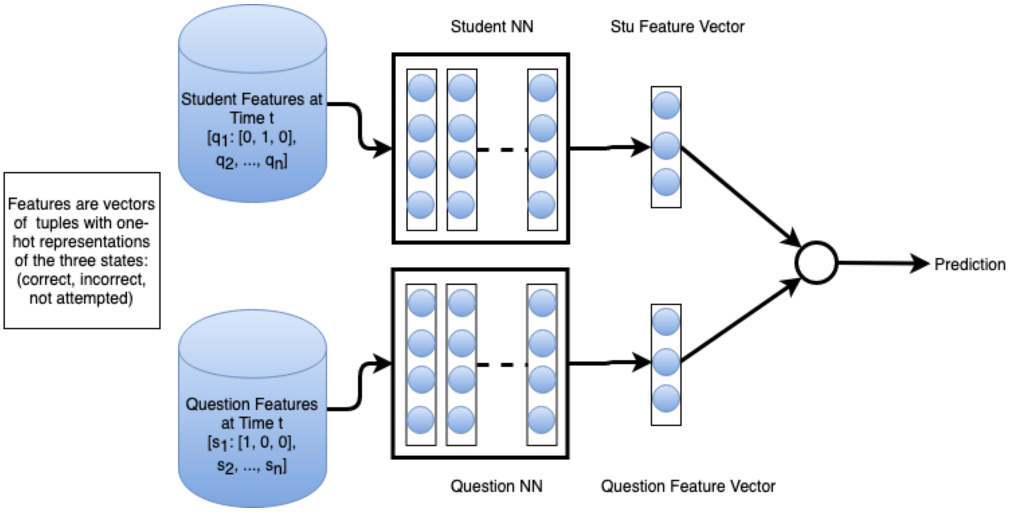

Student NN
Stu Feature Vector
Student Features at Time t [q1: [0, 1, 0], q2, ..., qn]
Features are vectors of tuples with one-hot representations of the three states: (correct, incorrect, not attempted)
Prediction
Question Features at Time t [s1: [1, 0, 0], s2, ..., sn]
Question NN
Question Feature Vector

Figure 3: Siamese-Like Neural Network

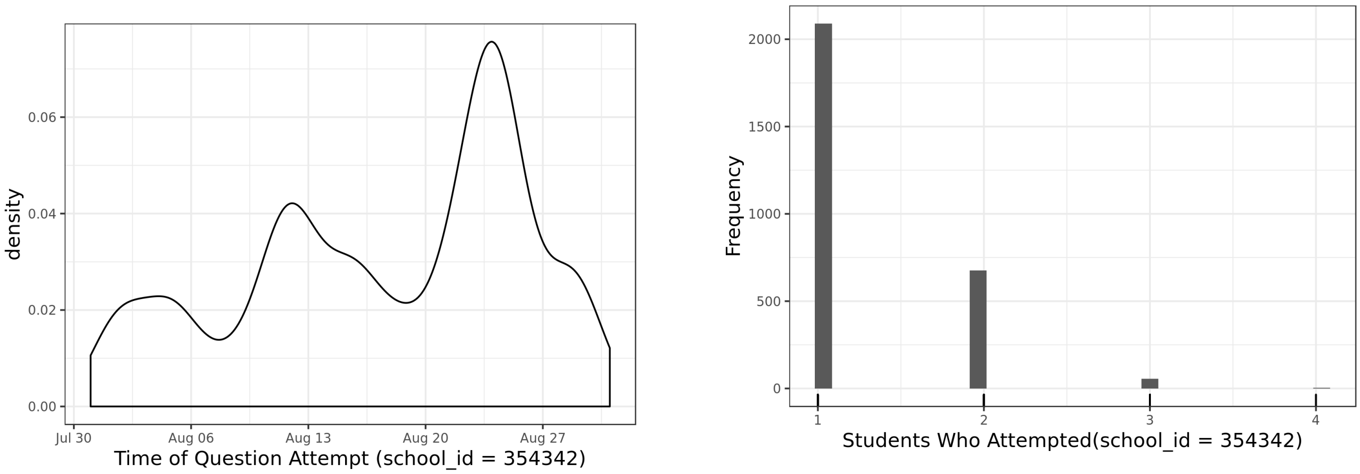
As a part of recent advancements in deep learning, Siamese networks have been responsible for the success of many one-shot learning tasks including near human-level performance on face recognition (Taigman et al., 2014). In the original paper by Bromley et al. (1993) on Siamese networks, the authors discussed how the structure of a pair of joining vectors lends itself to naturally rank similarities between inputs. The Siamese structure may lend itself to flexibly modeling ranking relationships between agent and object pairs for social science inquiries.

I propose using the model configuration described in Figure 3. I call this model Siamese-like neural network because different from the Siamese models proposed by Bromley et al. (1993), my model's parameters are not shared across students and questions.

This modeling approach is related to heterogeneous preference modeling. Recent efforts have been made by Athey et al. (2018) to estimate restaurant preferences using Bayesian latent parameter estimations to recover a matrix of consumer-restaurant preferences. In general, my Siamese network approach mirrors the matrix completion literature (Athey et al., 2017). I add the dimension of time to the targe matrix of student-question correctness probabilities by training a Siamese-like neural network on state-dependent representations of student and questions (see Data section for details).

3 Data

Data for this project comes from a mobile education company in China. The data contains logs of student questions responses on homework assigned by their school teacher. For the estimation, I use 138,001 records of student question responses from 105 student for 2777 unique questions from one school in one month in 2018.



(a) Density of Questions Answered over Time

(b) Distribution of Questions by Number of Students Who Attempted It

Before any machine learning algorithm can be run, it is essential to examine the structure of the dataset. The exposure of the same questions to different students is a pre-condition for applying the aforementioned methods. For accurate results, students would ideally be exposed to many questions over time. Figure 4a shows the number of questions answered by students in a particular school over time. Figure 4b show the distribution of questions by the number of students who have attempted it. We see that the majority of questions were attempted by one student but some where attempted by multiple students.

Unlike the naive Bayes models where the correctness is the only dimension of information I need, the Siamese-like NN needs features of questions as well as students. I represent any questions (j) at any point in time by a vector composed of a tuple for every individual student (i). Each tuple has 3 elements forming a one-hot representation of three states: 1) student i has not seen question j, 2) answered question j incorrectly, or 3) answered question j correctly. The individual at any point in time is represented by a vector composed of the same tuples for every individual question (j). This representation can be thought of as encapsulating the state-space where students and questions meet by vectors of tuples. My approach is similar in spirit to the Deep reinforcement learning literature (Mnih et al., 2013) where the value function is represented by a neural net over a large state space.

4 Results

4.1 Predictive Performance of Siamese-Like NN

The data ($138,001$ records) was randomly split into training (80%), development (10%) and test sets (10%). 50 sets of hyperparameters were randomly generated and models' F1 performance over the development set was used to select the final network structure documented in table 2. Table 1 documents the predictive accuracy of the best trained model on the test set. Figure 4.1 plots the ROC curve of the final Siamese-like NN model. We refrain from discussing the accuracy of the naive Bayes models as they are not designed to produce question-level predictions.

Model	Accuracy	Precision	Recall	F1 Score
Siamese-Like NN	0.873	0.907	0.937	0.922

Table 1: Siamese-Like NN Performance on the Test Set

Batch Size	Learning Rate	Epochs	Network Structure	Feature Vector Size
128	0.001	15	Student: [8331-400-80] Question: [315-200-80]	10

Table 2: Best Siamese-Like NN Structure

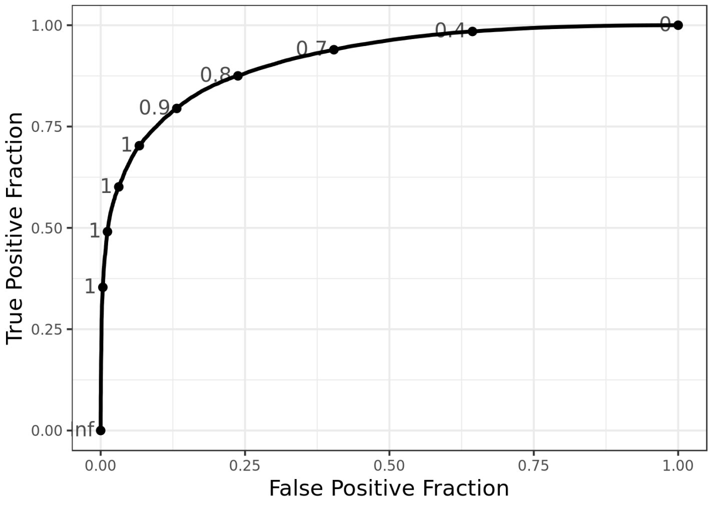


Figure 5: ROC Curves of Siamese-Like NN

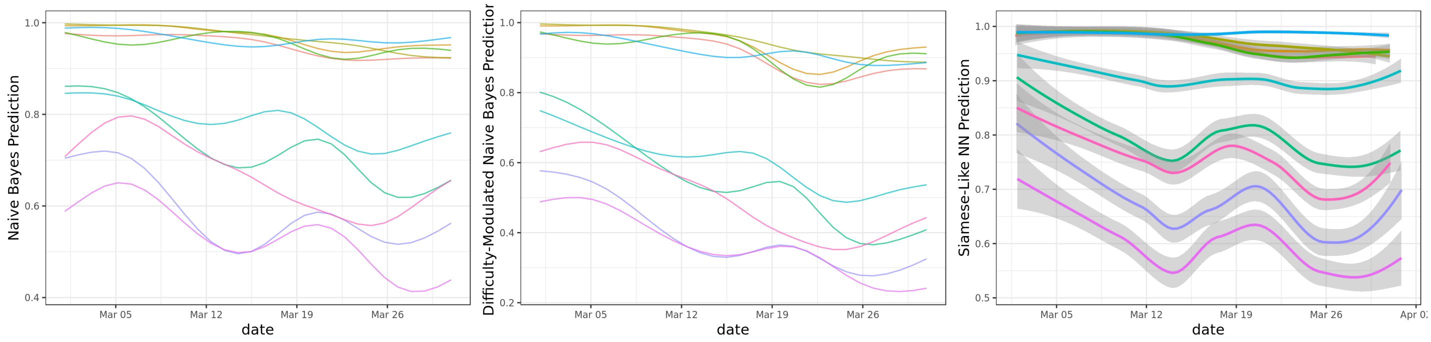


Figure 6: Prediction of Learning of the Three Models

Figure 6 shows the result of applying all three models to estimate student achievement (for a group of 10 students). As expected, the difficulty modulated naive Bayes model produced less fluctuations compared to the unmodulated naive Bayes model. Compared to the baseline models, the Siamese-like neural network results (smoothed over individual question predictions) produces more consistent predictions of achievement ranking between students over time. Coupled with its accuracy performance, this gives us more confidence that the Siamese-like NN is capturing variations in student achievement.

5 Conclusion and Future Work

My results suggest that the Siamese-like NN is capturing useful heterogeneity across both students and questions. To validate the usefulness of the Siamese-like NN over competing baseline models future work will need to link predictions with external measures of student learning. Another useful exercise is to explore the mechanisms underlying consistency of the rankings of the Siamese-like NN predictions.

In addition, many other models can be applied to the general task of recovering student achievement from student-question log data. Item-response theory model (EM Algorithm) from the education measurement literature may also be used to benchmark performance of more complex models. We can also reasonably expect lower rank matrix approximation (Athey et al., 2017) to be a competitive candidate.

References

Athey, S., Bayati, M., Doudchenko, N., Imbens, G., and Khosravi, K. (2017). Matrix Completion Methods for Causal Panel Data Models. *arXiv e-prints*, page arXiv:1710.10251.

Athey, S., Blei, D., Donnelly, R., Ruiz, F., and Schmidt, T. (2018). Estimating heterogeneous consumer preferences for restaurants and travel time using mobile location data. *arXiv preprint arXiv:1801.07826*.

Bromley, J., Guyon, I., Lecun, Y., Sackinger, E., and Shah, R. (1993). Signature verification using a siamese time delay neural network. In Cowan, J. and Tesauro, G., editors, *Advances in neural information processing systems (NIPS 1993)*, volume 6. Morgan Kaufmann.

McPherson, M. S. and Bacow, L. S. (2015). Online higher education: Beyond the hype cycle. *Journal of Economic Perspectives*, 29(4):135–54.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. A. (2013). Playing atari with deep reinforcement learning. *CoRR*, abs/1312.5602.

Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014). Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708.