Does student performance on prerequisite skills necessarily predict performance on postrequisite skills?

Background

Over the past several decades, educational software known as intelligent tutoring systems (ITS) have attempted to simulate the role of a human tutor with the goal of helping students on a large scale. As common with face-to-face tutoring, many ITS base their pedagogical design on skill building through scaffolding, practice, and mastery. ITS also support educators with dashboards that provide insights on student progress. These dashboards are a resource, which often can alert educators of at-risk learners. However, at the skill level, reporting is captured only after students have attempted to learn the skill in the system.

In this exploratory study, we use data from the web-based system ALEKS to fit a model that predicts when a student will struggle to master a skill before she is presented with the skill. We leverage the ITS’s knowledge map of pre- and postrequisite relationships among skills, as well its domain model, which requires students to master prerequisite skills before moving on to postrequisite skills.

Research Questions

- **RQ1:** Can we develop a model to effectively identify when a student will struggle on a postrequisite skill?
- **RQ2:** What features are most important in predicting success or fail on a postrequisite learning sequence?

Data and Method

- Learning sequences were gathered on both pre- and postrequisite items from the ALEKS course product Middle School Math Course 1 from 2017 to 2019. Row taken from original dataset
- Dataset consists of 235,772 students and 11,513,995 rows of data.
- There were 851 items represented in the dataset each having at least one direct prerequisite for which the student worked on.
- Review\(^2\) and various descriptive statistics were computed/ploted to help inform features.

- Our predictive model employed machine learning techniques seeking to predict when a student would produce a failed learning sequence.
- Descriptive statistics and previous work suggested the item and sequence length would be important features to consider.
- In all, 15 features were engineered from the original dataset. One of these is the postrequisite item, which resulted in a feature matrix with shape (11,513,995, 865).
- The feature matrix was separated into an 80% training and 20% test sets.
- The training set was trained on a 5-fold cross validation logistic regression using the scikit-learn LogisticRegressionCV classifier.

Results

- Performance was analyzed on the held out test set consisting of 20% of the overall dataset. The area under the receiver operating characteristic (ROC) curve was 0.794 ± 0.3, which meant that the model performed better than having no ability to distinguish pass vs. fail.
- The model showed difficulty identifying fails, as observed by the false positive group.
- We considered a more broad approach to see if our model was performing reasonably.
  
  a. False Positive: Why are these predicted to pass when in fact fail?
  b. False Negative: Why are these predicted to fail? Were they struggling to pass?

Discussion

- On average, the model tends to predict fail for large prerequisite sequences and pass for small prerequisite sequences for each of the four groups.
- **Re RQ1:** Focusing on the misclassified (false) groups, this trend reveals that performance on the prerequisite skill does not necessarily translate to the postrequisite skill, at least not with enough consistency for the model to predict when a struggling sequence will occur.

- **Re RQ2:** For model training efficiency, the feature data was stored as a sparse matrix, which makes it difficult to interpret the model’s coefficients. We can get a sense of the relative importance of each feature by using a technique called permutation feature importance, which randomly shuffles the values of a feature across all data points in the test set. This is done to one feature at a time leaving all other features untouched. The idea is, feature(s) that produce the largest change in the AUC are the most important. (These are shown in bold.)

Discussion

- Since performance on prerequisites does not predict performance on postrequisites with enough consistency, future work could investigate when it does and when it doesn’t. In the case of the false positives, perhaps there is a missing prerequisite skill needed, which could inform curriculum developers to add additional skills to the knowledge map.
- A limitation of the model is that the student is not a feature. Having a student feature could improve results considering it’s reasonable to suspect some students are more likely to struggle than others. While there is potential upside, this is likely difficult to implement in practice, as a model would have to update as the student continued to work in the system.

Acknowledgements

I would like to acknowledge and thank Jeffrey Matayoshi for his collaboration on this study and Shayan Doroudi for his feedback and guidance. I also want to thank Hannah Um for providing a de-identified dataset required for this study.

References