Analyzing Response Times and Answer Feedback Tags in an Adaptive Assessment

Jeffrey Matayoshi[0000−0003−1321−8159], Hasan Uzun, and Eric Cosyn

McGraw Hill ALEKS, Irvine, CA, USA
{jeffrey.matayoshi,hasan.uzun,eric.cosyn}@mheducation.com

Abstract. While many learning and assessment models focus on the binary correctness of student responses, previous studies have shown that having access to extra information—such as the time it takes students to respond to a question—can improve the performance of these models. As much of the previous work in this area has focused on knowledge tracing and next answer correctness, in this study we take a different approach and analyze the relationship between these extra types of information and the overall knowledge of the student, as measured by the end result of an adaptive assessment. In addition to looking at student response times, we investigate the benefit of having detailed information on the responses in the form of answer feedback tags from the adaptive assessment system. After using feature embeddings to encode the information from these feedback tags, we build several models and perform a feature importance analysis to compare the relative significance of these different variables. Although it appears that the response time variable does contain useful information, the answer feedback tags are ultimately much more important to the models.

Keywords: Adaptive assessment · Response times · Answer feedback

1 Introduction and Related Work

Of fundamental importance to learning and assessment models are the responses students give to the problems and questions they are asked. As these responses can vary widely in their content, a natural simplification is to summarize the responses by classifying them as being either correct or incorrect, a procedure currently used in many, if not most, existing student models [11,13]. However, many previous works have examined the use of other sources of information beyond these simple classifications. For example, some studies have shown that the use of student *response times*—i.e., the time a student takes when answering a question—can improve the accuracy of models for detecting engagement [1] or predicting student performance [7,16]. Additionally, other works have analyzed in detail the relationships between response times and student performance on either the next question [12,13] or the entire course [4].

Another goal of previous research has been to improve on the binary correct-or-incorrect classifications that are commonly used. Examples of this include assigning partial credit based on factors such as the number of attempts made or hints accessed [5,17,18], using a more granular classification scheme for the responses [13], or focusing on the specific information contained in wrong answers [12,14,19]. Lastly, other studies have used feature embeddings to capture the complex information contained in code submissions to programming questions [8,15], a relevant technique for our current study.

As much of this previous research focuses on knowledge tracing and next answer correctness, in this work we take a different approach and study the relationship between the overall knowledge of the student—as measured by the end result of an adaptive assessment—and the student responses. In addition to looking at the relationship between the response times of students and the results of the assessment, we investigate the potential for using more specific information about the student responses by leveraging the detailed answer feedback tags returned by the adaptive assessment. After using feature embeddings to encode the information from these feedback tags, we run a feature importance analysis to compare the relative significance of these different types of variables.

2 Background and Experimental Setup

Our study uses a data set obtained from ALEKS, an adaptive learning and assessment system. The ALEKS system contains an untimed, adaptive placement assessment that evaluates a student's mastery of 314 different *topics* from high school mathematics. Each assessment asks at most 30 questions, and the end result is a set of topics that the system believes the student knows. In what follows, we refer to the size of this set of topics as the *final score* of the student, with these scores ranging from a minimum of 0 to a maximum of 314.

During each assessment, an *extra problem* is randomly chosen from the 314 total topics and presented to the student—importantly, the student's response to this extra problem does not affect the final score of the assessment, and the data are instead used to evaluate the system. When presented a question during the assessment, a student can submit a response, which is then graded as correct or incorrect, or they can click on the "I don't know" (IDK) button if they are not familiar with the material. Since the majority of ALEKS topics require open-ended responses, the system uses a library of sophisticated algorithms to process these responses and determine if they are correct. In doing so, the system's algorithms return detailed information about the student responses in the form of *answer feedback tags*. These feedback tags might indicate that a student forgot to simplify their answer, or that they used the wrong unit of measurement.

The ALEKS placement assessment is used at a variety of community colleges and four-year institutions in the U.S., and the majority of the assessments are taken by students in their first year of school. For this study, we extract all available data for placement assessments taken over a time period starting in July 2016 and ending in November 2022, giving us 2,796,640 assessments from a total of 2,198,428 unique students—this amounts to roughly 1.3 assessments per student. We randomly partition the students into training, test, and validation sets of size 85% (1,868,664 students), 10% (219,843 students), and 5% (109,921 students), respectively. This results in a training set of 2,377,294 assessments, a test set of 279,369 assessments, and a validation set of 139,977 assessments.

3 Response Times

We define the response time to be the actual—or, real—time that elapses between the student's initial viewing of the extra problem and their submission of an answer, and we then compare these times to the final scores of the students. In our training data, the mean and median final scores are 148.5 and 140, respectively, with the first and third quartiles having values of 83 and 220, respectively. To normalize the response times, we follow the procedure outlined by Pelánek [12] and convert each response time into a percentile. Specifically, we partition all the extra problems based on the type of student response—correct, incorrect, or IDK—and the particular topic. Then, within the data for each response type and topic pair, we compute the percentiles for the response times to the extra problems. Finally, separately for each response type, we group the data points into bins of width one percentile, compute the average final score in each bin, and then plot the results in Figure 1.

While our focus is the overall knowledge of the students, in many ways the results are similar to those from previous studies on next answer correctness. For example, Figure 1 shows that students who spend the least amount of time on incorrect and IDK answers have the lowest average final scores, consistent with the results from other works that focused on wrong answers and their relationship with next answer correctness [12,13]. Additionally, our correct answer plot shows a decreasing trend, something that has been previously observed when studying the relationship between next answer correctness and correct responses to mathematics exercises [12,13].

4 Answer Feedback Tags

Our next analysis looks at the detailed answer feedback tags used by the ALEKS system to describe the student responses. Our data set contains a total of 31,954 unique feedback tags—an average of slightly more than 100 per topic. A small proportion of the responses (about 11%) do not have specific feedback tags, and to these we assign "generic" feedback tags that indicate if the response was classified as correct, incorrect, or IDK. To summarize the information from the feedback tags, we use the `Embedding` class from the PyTorch library [10]. Specifically, we train a neural network model in which each feedback tag is mapped to an n-dimensional vector containing unique information about the tag [6]. For this initial model, our only features are the feedback tag embeddings, while our target variable is the student's final score. Our neural network is a basic multilayer perceptron with two hidden layers of 10 hidden units each, and we use mean squared error (MSE) as our loss function. To more easily visualize the feedback tags, we use a 2-dimensional embedding—i.e., $n = 2$—for this initial model. The results are shown in Figure 2 where, to avoid obscuring the details, we restrict the plot to feedback tags with at least 500 data points, and we also exclude the generic feedback tags mentioned previously.

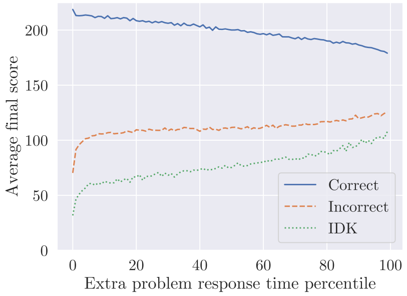

Fig. 1: Average final score versus extra problem response time percentile.

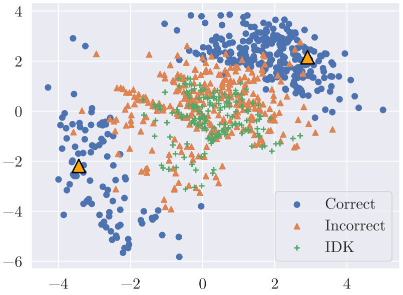

Fig. 2: Plot of 2-dimensional feedback embeddings.

While the neural network model seems to have mostly recovered the response classifications, as the correct feedback tags are fairly well-separated from the rest, there are some incorrect tags that are placed within large clusters of correct tags—two extreme cases are highlighted in Figure 2 with larger triangles. The highlighted tag in the lower left quadrant of Figure 2 appears if a student submits a value that properly satisfies a trigonometric equation, but is outside the range of the particular inverse trigonometric function being considered. For these responses, it seems likely the student has some experience with trigonometric functions, which are among the most advanced topics covered by the placement assessment. Notably, the average final score for these students in our training data is fairly high at 213.5 (N=590). The other highlighted feedback in Figure 2 indicates that a student, for the most part, correctly graphed a strict inequality, but made the small mistake of using a solid line instead of a dashed line. As before, the data indicate that students who receive this feedback tend to have a good understanding of the material in the placement assessment, as their average final score is a relatively high 212.2 (N=1119).

5 Feature Analysis

To evaluate the relative predictive strengths of the different features, we next train several simple neural network models, using the different combinations of features listed in the first column of Table 1. For the Time feature, rather than using the percentile scores we use the actual time taken by students to respond to the extra problem, as this gives better performance on our validation set. The Feedback feature uses the embeddings from the previous section, while the Response feature is a simplified version of the embedding model that encodes the response only as correct, incorrect, or IDK. For each set of features in the table, we apply a grid search and train 24 different models with various hyperparameters. Using root mean squared error (RMSE) as our chosen measure, from each set of 24 models we find the one that performs the best on our validation set, and we then evaluate this model on our test data. The resulting RMSE values, along with their confidence intervals,[1] are shown in the second column of Table 1. Notably, the models with the Feedback feature are more accurate than the models that use the simple classifications in the Response feature.

Table 1: Root mean squared error (RMSE) values on held-out test data. The feature importance RMSE values are averages computed from 10 iterations.

Model	**RMSE (95% CI)**	**Feature Importance**		
		Response	**Feedback**	**Time**
Response	69.3 (69.1, 69.4)	—	—	—
Response, Time	68.0 (67.8, 68.1)	95.1 (+27.1)	—	72.6 (+4.6)
Feedback	67.7 (67.6, 67.9)	—	—	—
Feedback, Time	66.6 (66.5, 66.8)	—	96.3 (+29.7)	70.5 (+3.9)

Finally, the right-hand side of Table 1 shows the results from an application of permutation feature importance [2], a method for quantifying the importance of the variables to the models. Since a higher RMSE value indicates a feature is more important, the Time feature is seemingly less important, as the average RMSE values are considerably lower when the Time feature is permuted—either 72.6 or 70.5—in comparison to the RMSE values when either the Response or Feedback values are permuted—95.1 or 96.3, respectively.

6 Discussion

Given that the ALEKS assessment measures a student's overall knowledge, it seems appropriate that the answer feedback tags, which potentially contain detailed information about a student's knowledge of a topic, are more important to the models than the response times. As response times do not directly measure the quality of the submitted answers, the use of response times could potentially lead to predictions that unfairly penalize students. For example, perhaps a student has a long response time because they are diligent and double-check their work; or, alternatively, a student may take longer to parse the instructions—possibly due to accessibility issues or struggles with reading comprehension—while being perfectly capable of performing the mathematical operations. Thus, for these reasons our future work is focused on potentially using the answer feedback tag feature embeddings, rather than the response times, to improve the ALEKS system's recently introduced neural network assessment engine [9].

[1] To account for students with multiple assessments, the confidence intervals are computed using the cluster bootstrap method [3].

References

1. Beck, J.E.: Engagement tracing: Using response times to model student disengagement. In: Artificial Intelligence in Education (2005)
2. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001)
3. Field, C.A., Welsh, A.H.: Bootstrapping clustered data. Journal of the Royal Statistical Society: Series B (Statistical Methodology) **69**(3), 369–390 (2007)
4. González-Espada, W.J., Bullock, D.W.: Innovative applications of classroom response systems: Investigating students item response times in relation to final course grade, gender, general point average, and high school ACT scores. Electronic Journal for the Integration of Technology in Education **6**, 97–108 (2007)
5. Inwegen, E.V., Adjei, S.A., Wang, Y., Heffernan, N.T.: Using partial credit and response history to model user knowledge. In: Educational Data Mining (2015)
6. Jurafsky, D., Martin, J.H.: Speech and Language Processing (3rd ed. draft) (2021), https://web.stanford.edu/~jurafsky/slp3/
7. Lin, C., Shen, S., Chi, M.: Incorporating student response time and tutor instructional interventions into student modeling. In: User Modeling Adaptation and Personalization (2016)
8. Liu, N., Wang, Z., Baraniuk, R.G., Lan, A.: Open-ended knowledge tracing (2022). https://doi.org/10.48550/ARXIV.2203.03716, https://arxiv.org/abs/2203.03716
9. Matayoshi, J., Uzun, H., Cosyn, E.: Using a randomized experiment to compare the performance of two adaptive assessment engines. In: Educational Data Mining. pp. 821–827 (2022)
10. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E.Z., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S.: PyTorch: An imperative style, high-performance deep learning library. CoRR **abs/1912.01703** (2019), http://arxiv.org/abs/1912.01703
11. Pelánek, R.: Bayesian knowledge tracing, logistic models, and beyond: An overview of learner modeling techniques. User Modeling and User-Adapted Interaction **27**(3), 313–350 (2017)
12. Pelánek, R.: Exploring the utility of response times and wrong answers for adaptive learning. In: Learning @ Scale. pp. 1–4 (2018)
13. Pelánek, R., Effenberger, T.: Beyond binary correctness: Classification of students' answers in learning systems. User Modeling and User-Adapted Interaction **30**(5), 867–893 (2020)
14. Pelánek, R., Rihák, J.: Properties and applications of wrong answers in online educational systems. In: Educational Data Mining (2016)
15. Piech, C., Huang, J., Nguyen, A., Phulsuksombati, M., Sahami, M., Guibas, L.: Learning program embeddings to propagate feedback on student code. In: International Conference on Machine Learning. pp. 1093–1102. PMLR (2015)
16. Wang, Y., Heffernan, N.T.: Leveraging first response time into the knowledge tracing model. In: Educational Data Mining (2012)
17. Wang, Y., Heffernan, N.T.: Extending knowledge tracing to allow partial credit: Using continuous versus binary nodes. In: Artificial Intelligence in Education (2013)
18. Wang, Y., Heffernan, N.T., Beck, J.E.: Representing student performance with partial credit. In: Educational Data Mining (2010)
19. Wang, Y., Heffernan, N.T., Heffernan, C.: Towards better affect detectors: Effect of missing skills, class features and common wrong answers. Learning Analytics and Knowledge (2015)