Using a Randomized Experiment to Compare the Performance of Two Adaptive Assessment Engines

Jeffrey Matayoshi
McGraw Hill ALEKS
jeffrey.matayoshi
@mheducation.com

Hasan Uzun
McGraw Hill ALEKS
hasan.uzun
@mheducation.com

Eric Cosyn
McGraw Hill ALEKS
eric.cosyn
@mheducation.com

ABSTRACT

Knowledge space theory (KST) is a mathematical framework for modeling and assessing student knowledge. While KST has successfully served as the foundation of several learning systems, recent advancements in machine learning provide an opportunity to improve on purely KST-based approaches to assessing student knowledge. As such, in this work we compare the performance of an existing KST-based adaptive assessment to that of a newly developed version—with this new version combining the predictive power of a neural network model with the strengths of existing KST-based approaches. Using a cluster randomized experiment containing data from approximately 140,000 assessments, we show that the new neural network assessment engine improves on the performance of the existing KST version, both on standard classification metrics, as well as on measures more specific to the student learning experience.

Keywords

Adaptive assessment, neural networks, knowledge space theory

1. INTRODUCTION

Combining elements of probability theory and combinatorics, knowledge space theory (KST) is a mathematical approach to the modeling and assessing of student knowledge [12, 15]. KST-based adaptive assessments focus on identifying the set of topics that are most likely to be known by a student, with many such implementations having been developed over the years [6, 7, 18, 20]. The particular system at the center of this work is ALEKS, an adaptive learning system covering subjects such as math, statistics, and chemistry. The foundation of the system is an *initial assessment* that is given to a student at the beginning of an ALEKS course, with the information from this assessment being used to guide their subsequent learning in the system. The current version of the initial assessment engine uses a KST-based model that employs many features and concepts described in the KST literature (informative summaries of these ideas can be found in [13] and Chapter 13 of [15]).

Several works have evaluated the validity and reliability of KST-based assessments, with the results indicating that such assessments are both accurate and valid [5, 6, 10, 28]. However, current innovations in machine learning and, in particular, neural network models provide an opportunity to improve on purely KST-based approaches to adaptive assessments. As such, a newer version of the ALEKS initial assessment was recently developed and released into production for testing. By augmenting the KST-based adaptive assessment approach with a neural network model that can better leverage the years of accumulated ALEKS user data, we hope to make improvements to the accuracy and efficiency of the initial assessment. Thus, in the current study we evaluate this new assessment engine by analyzing the results from a randomized experiment—or, A/B test—comparing the performance of the two different versions of the ALEKS initial assessment.

2. BACKGROUND

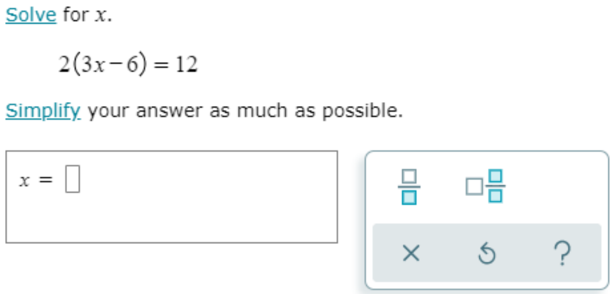
In this section, we give a brief background of the ALEKS system and the two versions of its initial assessment engine. Within the ALEKS system, a *topic* is a problem type that covers a discrete unit of an academic course—Figure 1 contains a screen capture of an example math topic titled "Introduction to solving an equation with parentheses." A *knowledge state* is a collection of topics that, conceivably, a student can know at any one time. The collection of knowledge states is known as the *knowledge space*. Based on the knowledge space, the topics in an ALEKS course contain many *prerequisite* relationships. That is, topic a is a prerequisite for topic b if a contains certain core material that must be mastered before learning the material in b.

In order to ensure students are learning the most appropriate topics, the initial assessment is given at the start of an ALEKS course, with the purpose of this assessment being to measure the student's incoming knowledge. In this assessment, a student is presented topics from the course, and for each topic they can either submit a response—which is then graded as either correct or incorrect—or they can click on the "I don't know" button if they are unable to answer the question. This assessment is adaptive, in that it selects the current question based on the responses to the earlier questions in the assessment. At most 30 questions are asked, which balances the need to acquire information on the student's knowledge state with the possibility of overwhelming the student.[1] After each question, a probability is computed for each topic in the course, with this probability estimating how likely it is that the student knows the topic. At the end of the assessment, based on both these probability estimates and the prerequisite relationships in the knowledge space, the ALEKS system partitions the topics in the course into the following three categories.

- Topics that are most likely known
- Topics that are most likely unknown
- All remaining topics (uncertain)

The student's knowledge state consists of the topics in the known category.[2] Due to the enormous numbers of possible knowledge states, KST-based assessments typically use simplifications in order to model the relationships between topics. Perhaps the most common approach is to focus solely on identifying the prerequisite—or, topic to topic—relationships when developing the knowledge space; these are variously known as partial order or ordinal knowledge spaces [8, 9, 11, 15]. While adaptive assessments using these approaches have been successful [6, 7, 10, 20], there is an inherent loss of information when mainly focusing on the pairwise relationships between the topics. Although methods have been outlined for moving beyond pairwise relationships and looking at larger groups of topics simultaneously, the computational complexity grows substantially with each added topic, and research has thus far been mostly limited to small groups of at most three or four topics [11].

Motivated by this issue, the new version of the ALEKS assessment engine is powered by a neural network model that, for each topic, estimates the probability of a correct answer. The advantage of a neural network is that it has the flexibility to model more complex relationships between the topics—that is, it can go beyond focusing on specific relationships between pairs or very small groups of topics. However, for both practical and theoretical reasons, it's desirable that the neural network model work within the existing KST framework of the ALEKS system. While other attempts have been made at applying neural networks to ALEKS data [21, 22, 24], none of these previous models took into account the specific details of the knowledge spaces. As such, the distinguishing feature of the neural network used for the initial assessment is that it applies a specially designed architecture to output probabilities consistent with the knowledge space—in particular, this architecture ensures that the probability estimates follow the prerequisite relationships among the topics. Thus, the neural network can leverage the vast amounts of ALEKS data to make accurate predictions, while simultaneously respecting the set of knowledge states and thereby ensuring these predictions are pedagogically sound. Notably, similar ideas have been successfully applied in knowledge tracing models, where it's been shown that leveraging prerequisite relationships can be done effectively through the loss function of a neural network [2], or by utilizing dynamic Bayesian networks [19].



Solve for x.

$$2(3x-6)=12$$

Simplify your answer as much as possible.

x = □

Figure 1: Screen capture of the ALEKS topic "Introduction to solving an equation with parentheses."

3. EXPERIMENTAL SETUP

Neural network models were trained for eight ALEKS products: middle school math courses 1-3 (in the U.S., these correspond to grades 6 through 8); general chemistry A and B (the first and second semesters of college-level general chemistry, respectively); and three college-level math classes (precalculus, college algebra, and college algebra with trigonometry). Although we do not have access to specific demographic information for the students in our data, we can say that the majority of the middle school users are from U.S. public schools, while the higher education users come from both community colleges and four-year institutions, again mainly from the U.S. We began evaluating the new assessment engine in mid-December of 2021, using A/B testing to compare its performance to that of the existing version of the assessment, with data being collected through February of 2022. For the aforementioned products, we used a cluster randomized design to assign the two versions of the assessment at the class level, an approach commonly used for educational studies [14, 27, 29]. Specifically, each class was randomly assigned to receive either the new assessment or the old assessment, whereupon all the students in that class then received the same version of the assessment.

For the middle school and chemistry products, the neural network models were trained with the exact data sets that were used to build the models for the existing version of the ALEKS assessment engine; as such, our analyses on these products are the most informative when trying to precisely estimate the differences in performance between the two engines. In comparison, rather than being restricted to a certain time period, the college math neural network models were trained on all the available data. Thus, from the purely scientific perspective of measuring the differences between the assessment engines, comparing the results on the college math courses is perhaps less useful. However, from the more practical standpoint of understanding the (possible) gains from updating existing ALEKS products with the new engine, we believe this analysis to be informative. That is, there are currently many older ALEKS products that haven't been updated over the past several years; thus, if these products were updated with the new neural network assessment engine, they would stand to gain from both the application of the neural network *and* the extra data used to train the model. As such, we believe this analysis gives some insight into the potential benefits from converting these older ALEKS products to the new assessment engine.

[1] See [23] for evidence of a 'fatigue effect' experienced by students towards the end of an ALEKS assessment.

[2] The distinction between the unknown and uncertain topics is mainly relevant for the student's learning in the system, with these categorizations determining the amount of work required before a topic is considered to be mastered.

Table 1: Summary statistics for the three groups of ALEKS products.

Product	Type	Total Classes	Total Assessments	Average number of topics	Extra problem correct rate
Middle school math	KST	5,311	19,480	433.6	0.334
	NN	5,407	19,313	436.4	0.330
Chemistry	KST	360	16,315	158.0	0.264
	NN	381	18,812	155.2	0.257
College math	KST	1,247	33,207	258.2	0.317
	NN	1,202	30,785	253.9	0.311

4. ANALYSIS

In order to compare the performance of the two assessment models, in what follows we make use of an *extra problem* that is asked during each initial assessment. This extra problem is chosen uniformly at random from the topics in the course and presented to the student as an assessment question. However, the answer to the extra problem does not affect the results of the assessment—instead, the information from the extra problem is used to evaluate and improve the ALEKS system. In Table 1 we show summary statistics after partitioning the products into three distinct groups—middle school math (middle school math courses 1–3), chemistry (general chemistry A and B), and college math (precalculus, college algebra, and college algebra with trigonometry)—and also the two treatment arms of our analysis—the KST-based assessment, and the neural network assessment (NN). In addition to showing the total numbers of classes and initial assessments taken, we also show the average number of topics used by each class, weighted by the number of assessments per class.[3] Finally, in the last column we display the average correct answer rate to the extra problem.

Our first analysis compares the performance of the two assessment models by treating them as binary classifiers, where we consider a positive outcome to be a correct answer to the extra problem, while a negative outcome is either an incorrect or "I don't know" response. We compare the assessment engines using three metrics: area under the receiver operating characteristic curve (AUROC), point biserial correlation (r_{pb}), and accuracy score. AUROC is frequently used to evaluate probabilistic classifiers, and it is known to perform well even with some class imbalance [16]. The point biserial correlation is a special case of the Pearson correlation coefficient in which one variable is dichotomous (the student response) and the other variable is continuous (the probability estimate from the assessment).[4] While the actual probabilities are used in the computations of AUROC and the point biserial correlation, for the accuracy calculation we assume any probability at or above 0.5 is a positive prediction, with anything below 0.5 then being considered a negative prediction. We should clarify that while this assignment of prediction labels is a standard procedure used to evaluate binary classifiers, it does not necessarily correspond to the actual classifications made by the ALEKS system—we look at these ALEKS-specific classifications in more detail later.

As students are grouped—or, "clustered"—into classes, to compute confidence intervals around the point estimates we use the following bootstrap procedure, applied separately to each product and assessment engine pair. That is, we apply this procedure a total of six times: once for the middle school products using the KST engine, then for the middle school products using the neural network engine, then for the chemistry products with the KST engine, etc. The first step in the procedure is to resample our data using the *cluster bootstrap*, a modified version of the standard bootstrap that specifically works with clustered data [17]. For our analysis, classes are randomly sampled with replacement from our original data set, until we have a sample of classes equal to the number in our original data set. Then, we combine all the assessments from these selected classes to generate one bootstrap sample—note this means that some classes, as well as the associated assessments, appear multiple times in the sample. Next, we compute our statistics for this bootstrap sample, and we then repeat this entire procedure until we've generated 20,000 bootstrap samples in total. Finally, since each resulting bootstrap distribution turns out to be symmetric and centered at the original values of the statistic—i.e., the value of the statistic computed from our original set of data—we compute each confidence interval by simply taking the 2.5th and 97.5th percentiles.

The results are shown in Table 2. For the middle school products, we can see that the neural network engine performs better according to each of the metrics, even after taking into account the confidence intervals. Next, the results for the chemistry products are less conclusive. Although the point estimates are all higher for the neural network engine, based on the confidence intervals in the third column there's uncertainty with the signs of these differences. Finally, for the college math products the neural network engine again has much stronger performance compared to the KST-based assessment—we reiterate that this is expected, as the college math neural network models had access to more recent data in comparison to the corresponding KST models.

While the results for the middle school and college math

[3] While each course has a default set of recommended topics, the instructor is free to add or remove topics from this set.

[4] The Matthews correlation coefficient (MCC) [25] is a related statistic that is regarded as being an informative measure for evaluating binary classifiers [1, 3, 4, 26]. As the MCC is mathematically equivalent to the Pearson correlation coefficient of two dichotomous variables—also known as the phi coefficient—the only difference from computing the point biserial correlation is that the MCC requires we dichotimize the probability estimates. However, since this would result in some loss of information, we prefer to use the point biserial correlation for our current evaluations.

Table 2: Comparison of the two assessment engines, using the area under the receiver operating characteristic curve (AUROC), point biserial correlation coefficient (r_{pb}), and accuracy score. Numbers in parentheses represent the 95% confidence intervals computed from 20,000 cluster bootstrap samples.

Product	Metric	Assessment type		Difference
		KST	NN	NN−KST
Middle school math	AUROC	0.875 (0.870, 0.881)	0.894 (0.889, 0.899)	0.019 (0.011, 0.026)
	r_{pb}	0.624 (0.612, 0.635)	0.674 (0.664, 0.684)	0.050 (0.035, 0.065)
	Accuracy	0.811 (0.805, 0.817)	0.831 (0.825, 0.836)	0.020 (0.012, 0.028)
Chemistry	AUROC	0.871 (0.861, 0.881)	0.880 (0.873, 0.888)	0.009 (-0.003, 0.022)
	r_{pb}	0.610 (0.585, 0.634)	0.629 (0.613, 0.646)	0.019 (-0.010, 0.049)
	Accuracy	0.837 (0.829, 0.845)	0.838 (0.827, 0.849)	0.001 (-0.013, 0.014)
College math	AUROC	0.861 (0.856, 0.867)	0.898 (0.893, 0.902)	0.036 (0.029, 0.043)
	r_{pb}	0.599 (0.589, 0.610)	0.674 (0.664, 0.684)	0.075 (0.060, 0.089)
	Accuracy	0.814 (0.807, 0.821)	0.838 (0.833, 0.843)	0.024 (0.015, 0.032)

products are encouraging, the performance of the chemistry neural network models is slightly unexpected. Based on our previous evaluations when training the neural network models, we expected a larger performance improvement over the KST-based assessment for the chemistry products—in comparison, the performance of the middle school and college math neural network assessments are consistent with our expectations. A possible concern is that there are differences between the chemistry student populations using the two different assessment engines—while not conclusive, some of the statistics in Table 1 are suggestive of such a difference. To start, although there are over 35,000 total assessments taken for the chemistry products, the number of unique classes is low in comparison to the other products—for example, the number of chemistry classes (741) is a small fraction of the number of middle school classes (10,718). This is important, since in a cluster randomized experiment such as ours, the number of clusters is typically more restrictive than the overall sample size [29], and such designs have a higher risk of non-equivalence between the experimental groups [14, 27]. Additionally, for the chemistry products the neural network group has about 15% more assessments taken than the KST group, which is possibly another sign of non-equivalence between the groups. While these differences aren't conclusive, they at least suggest that the student populations may differ in some respect. Thus, in the next section we analyze the chemistry products further in order to obtain a better understanding of the results.

5. REANALYZING THE CHEMISTRY ASSESSMENTS

To investigate the possibility that the student populations are not equalized across the chemistry experimental groups, we take advantage of the fact that an ALEKS assessment can be "replayed" on an assessment engine different from the one that was originally used. For example, suppose a student takes an assessment using the KST-based engine. Once this assessment is completed, we can feed the questions and responses to the neural network engine—taking care to remove the extra problem from this process—generate probability estimates, and then evaluate the probability for the extra problem in the original assessment. The main drawback to this approach is that the engine used for the replay assessment won't be able to choose the questions that are given to the student, which could theoretically bias the results somewhat. However, the advantage of this approach, and the reason we employ it here, is that it allows us to directly compare the assessment engines on the same sets of data, removing any concerns about the non-equivalence of the experimental groups.

To that end, using the data from the 16,315 chemistry assessments originally taken with the KST engine, we feed the questions and responses to the neural network models and generate probability estimates. We then take these probabilities and compute our evaluation metrics on the extra problems. Next, we repeat the same procedure in the other direction—that is, using the data from the 18,812 chemistry assessments that originally used the neural network engine, we take the questions and responses from each assessment and feed them to the KST-based engine. As before, we use the resulting probabilities to compute our evaluation metrics on the extra problems.

Table 3: Comparison of the replayed assessments on the chemistry products. Numbers in parentheses represent the 95% confidence intervals computed from 20,000 cluster bootstrap samples.

Metric	Assessment type		Difference
	KST	NN	NN - KST
AUROC	0.856 (0.847, 0.866)	0.889 (0.881, 0.898)	0.033 (0.020, 0.046)
r_{pb}	0.581 (0.561, 0.602)	0.651 (0.630, 0.671)	0.070 (0.041, 0.099)
Accuracy	0.824 (0.812, 0.835)	0.844 (0.836, 0.853)	0.021 (0.007, 0.035)

The results are shown in Table 3, where we can see a large difference in performance between the two assessment engines. In contrast to the results from Table 2, on the replayed assessments the neural network engine does much better, while the performance of the KST assessment engine has dropped noticeably. As such, the contrast in performance between the assessment engines is clearer, with the 95% confidence intervals for the differences (third column) all bounded away from zero. So, it does appear likely that there are underlying population differences between the groups of students using the two assessment engines. Thus, arguably the fairest comparison is to use both the actual assessment data and the replayed assessment data, and then compare the assessment engines based on this combined data set. The results are shown in Table 4, where we can see a relatively clear performance gap between the two engines, albeit not quite as large as in Table 3.

Table 4: Combined comparison of the two assessment engines on the chemistry products, including data from both the original and replayed assessments. Numbers in parentheses represent the 95% confidence intervals computed from 20,000 cluster bootstrap samples.

Metric	Assessment type		Difference
	KST	NN	NN - KST
AUROC	0.863 (0.856, 0.870)	0.885 (0.879, 0.891)	0.022 (0.012, 0.031)
r_{pb}	0.594 (0.578, 0.610)	0.639 (0.626, 0.653)	0.045 (0.024, 0.067)
Accuracy	0.830 (0.823, 0.837)	0.841 (0.834, 0.848)	0.011 (0.001, 0.021)

6. ALEKS KNOWLEDGE STATES

While the previous analyses have compared the performance of the assessment engines assuming they are standard binary classifiers, in this section we compare the models based on measures more specific to the ALEKS system. In what follows, we restrict our analyses to the middle school data, as we believe this data set gives the most balanced and fair comparison between the two assessment engines.

Recall that the purpose of the initial assessment is to identify the topics in a student's knowledge state—that is, the topics in the known category. Since the ALEKS system uses this information to determine what a student is ready to learn, inaccurately measuring a knowledge state would negatively affect the student's learning experience. For example, giving a student credit for many topics that aren't justified could cause the student to start their learning with topics for which they aren't prepared, possibly leading to frustration. On the other hand, not giving a student enough credit for topics they know has an opposite effect, as the student may start with topics that are too easy, causing boredom.

To start, in Figure 2 we show a relative frequency histogram of the number of topics classified as known after each assessment is completed. The striped (blue) bars represent the 19,313 assessments from the neural network engine, while the solid bars show the proportions for the 19,480 assessments from the KST engine. The mean and median are 112.5 and 98, respectively, for the neural network engine; in comparison, these values are 106.9 and 85 for the KST engine. Depending on whether we use the mean or median to describe the outcome of a "typical" assessment, the knowledge states from the neural network assessment engine are larger by either 5.2% (mean) or 15.3% (median). Furthermore, the first and third quartiles for the neural network engine are 52 and 159, respectively, with these values being 42 and 152 for the KST engine. These differences possibly indicate that the advantages of the neural network engine apply to a diverse sample of students, rather than only those in a specific part of the distribution. Moreover, it's encouraging that the gains are larger for the students in the first quartile, as the relative benefit of the additional topics is greater for students with smaller knowledge states.

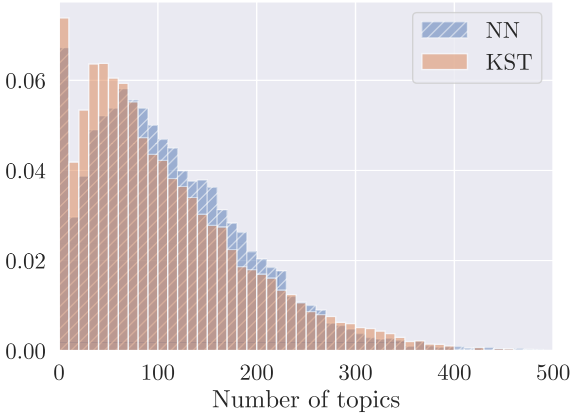


Figure 2: Relative frequency histogram of the number of known topics for the middle school products.

While the knowledge states from the neural network engine tend to be larger, it's important that they are also accurate; assigning more topics to the known category is of limited use if the topics aren't actually known by the students. As such, in our next analysis we look at how often students answer correctly to the extra problem based on the classification (or, categorization) of the ALEKS system—either known, unknown, or uncertain. The results are shown in Table 5. Starting with the known category, we can see that students answer correctly more often with the neural network engine compared to the KST engine—0.792 vs. 0.788. Additionally, the neural network classifies the extra problem as being known more often than the KST model—0.271 vs. 0.259—which is consistent with the results in Figure 2. Next, note that while we want a high rate of correct answers to the topics in the known category, for topics in the unknown category we want the opposite—that is, a low correct answer rate to the unknown topics indicates the classifications are accurate. We can see that the unknown topics for the neural network engine have a lower correct answer rate in comparison to those from the KST engine—at the same time, the proportions of topics labeled as unknown are comparable. Finally, the neural network engine has a lower proportion of topics in the uncertain category, showing that overall it's more aggressive in labeling topics as known or unknown.

In this last analysis we'd like to get a different perspective on the performance of the models. In particular, both assessment engines rely on the same underlying knowledge spaces, which means the prerequisite relationships between the topics are the same. Furthermore, when a topic is answered correctly during an assessment, the ALEKS system uses these prerequisite relationships to classify topics as being known—specifically, if a topic is answered correctly, that topic, as well as all of its prerequisites, are classified as known. Since

Table 5: Statistics for the assessment engines, partitioned by the classification of the extra problem.

Classification		Assessment type	
		KST	NN
Known	Proportion	0.259	0.271
	Correct rate	0.788	0.792
Unknown	Proportion	0.573	0.574
	Correct rate	0.106	0.088
Uncertain	Proportion	0.168	0.154
	Correct rate	0.412	0.421

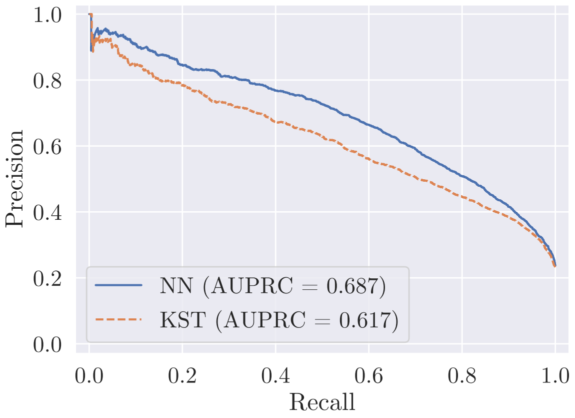


Figure 3: Precision-recall curves for the extra problems that are not directly classified by the prerequisite relationships.

this behavior is the same for each assessment engine, we want to remove these classifications and obtain a more nuanced view of the differences in performance. Additionally, these "prerequisite" classifications tend to be the easiest ones for the assessment engines to make, and so by removing them we can evaluate the engines on a more challenging subset of the data. To that end, we remove any extra problem that (a) also appeared as a regular assessment question and was answered correctly, or (b) is a prerequisite of a topic that was correctly answered during the assessment—this leaves us with 16,122 and 15,963 extra problems for the KST and neural network engines, respectively. Since the performance of the ALEKS system depends on the assessment's ability to correctly classify the topics in a student's knowledge state (i.e., precision), while simultaneously identifying as many such topics as possible (i.e., recall), for these data points we plot the precision-recall curves, as we can then compare the performance of the assessment engines across a range of thresholds. The results are shown in Figure 3, where we can see that the neural network curve dominates for the vast majority of the recall values, with the precision values being substantially higher in many places. Overall, it's informative to see the strong performance of the neural network assessment engine on this specific subset of the data.

7. DISCUSSION

In this work we presented and analyzed the results of a randomized experiment—or, A/B test—comparing two different versions of the ALEKS initial assessment engine. The purpose of this analysis was to validate the neural network assessment engine and verify it improves upon the existing version that's based on knowledge space theory (KST). While an initial analysis indicated strong improvement for the middle school and college math products, the difference was less clear for the chemistry products. To investigate further, we reanalyzed the chemistry assessments by "replaying" each of them with the competing engine. This analysis suggested that the student populations in the two experimental groups for the chemistry products were not completely equivalent, confounding the comparison. Thus, we adjusted for the non-equivalence of the student populations by combining the data from both the replayed and actual assessments, with the results indicating that the neural network assessment outperformed the KST-based version. Finally, we evaluated the assessment engines on metrics more specific to the ALEKS system—in these analyses, we saw that the neural network assessment gave students credit for knowing more topics, while simultaneously being more accurate.

The chemistry results were interesting and somewhat surprising. Given the large numbers of students in the two experimental groups, it was unexpected that, as suggested by the replay results, these groups would be dissimilar. However, since many of the chemistry students are from large universities, the class sizes also tend to be large—thus, as seen in Table 1 the number of distinct classes is relatively small. (In comparison, the middle school products have a higher number of "independent" users—e.g., homeschooling/home education students, or individual students seeking extra help—resulting in smaller average class sizes.) In experimental designs with multilevel structure such as ours, the number of clusters—represented by the student classes in our data—is typically more important than the overall sample size in ensuring the baseline equivalence of the experimental groups [14, 27, 29]. As such, our experience highlights the fact that the use of cluster randomized designs, while desirable in education research for several reasons, can lead to difficulties with the statistical analysis of the results, and that this can be an issue even with seemingly large samples of data.

Overall, we found the performance of the neural network assessment engine to be promising, in that it has the potential to benefit students in multiple ways. For example, the fact that it returns larger knowledge states in comparison to the KST engine—and, importantly, without a drop in accuracy—means that students do not have to spend as much time working on topics they may already know. Thus, students can learn more efficiently and spend more time working on completely new material, hopefully allowing them to progress further in the course. Yet another possible benefit pertains directly to the initial assessment itself. User feedback from both students and teachers has informed us that there is a desire for a shorter initial assessment, one that asks fewer questions and takes less time to finish. Given the gain in performance, it seems plausible that the neural network assessment could still improve on the KST-based assessment even if fewer questions are asked. We are currently looking at this possibility in detail, with the hope of shortening the initial assessment and improving the student experience within the ALEKS system.

8. REFERENCES

[1] S. Boughorbel, F. Jarray, and M. El-Anbari. Optimal classifier for imbalanced data using Matthews correlation coefficient metric. *PloS one*, 12(6):e0177678, 2017.

[2] P. Chen, Y. Lu, V. W. Zheng, and Y. Pian. Prerequisite-driven deep knowledge tracing. In *2018 IEEE International Conference on Data Mining (ICDM)*, pages 39–48. IEEE, 2018.

[3] D. Chicco. Ten quick tips for machine learning in computational biology. *BioData mining*, 10(1):1–17, 2017.

[4] D. Chicco and G. Jurman. The advantages of the matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13, 2020.

[5] E. Cosyn, C. Doble, J.-C. Falmagne, A. Lenoble, N. Thiéry, and H. Uzun. Assessing mathematical knowledge in a learning space. In J.-C. Falmagne, D. Albert, C. Doble, D. Eppstein, and X. Hu, editors, *Knowledge Spaces: Applications in Education*, pages 27–50. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[6] E. Cosyn, H. Uzun, C. Doble, and J. Matayoshi. A practical perspective on knowledge space theory: ALEKS and its data. *Journal of Mathematical Psychology*, 101:102512, 2021.

[7] D. de Chiusole, L. Stefanutti, P. Anselmi, and E. Robusto. Stat-Knowlab. Assessment and learning of statistics with competence-based knowledge space theory. *International Journal of Artificial Intelligence in Education*, 30(4):668–700, 2020.

[8] M. C. Desmarais and M. Gagnon. Bayesian student models based on item to item knowledge structures. In *European Conference on Technology Enhanced Learning*, pages 111–124. Springer, 2006.

[9] M. C. Desmarais, A. Maluf, and J. Liu. User-expertise modeling with empirically derived probabilistic implication networks. *User modeling and user-adapted interaction*, 5(3):283–315, 1995.

[10] C. Doble, J. Matayoshi, E. Cosyn, H. Uzun, and A. Karami. A data-based simulation study of reliability for an adaptive assessment based on knowledge space theory. *International Journal of Artificial Intelligence in Education*, 29(2):258–282, 2019.

[11] J.-P. Doignon and J. Falmagne. Knowledge spaces and learning spaces. *arXiv preprint arXiv:1511.06757*, pages 1–51, 2015.

[12] J.-P. Doignon and J.-C. Falmagne. Spaces for the assessment of knowledge. *International Journal of Man-Machine Studies*, 23:175–196, 1985.

[13] J.-P. Doignon, J.-C. Falmagne, and E. Cosyn. Learning spaces: A mathematical compendium. In J.-C. Falmagne, D. Albert, C. Doble, D. Eppstein, and X. Hu, editors, *Knowledge Spaces: Applications in Education*, pages 131–145. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

[14] J. Dreyhaupt, B. Mayer, O. Keis, W. Öchsner, and R. Muche. Cluster-randomized studies in educational research: principles and methodological aspects. *GMS Journal for Medical Education*, 34(2), 2017.

[15] J.-C. Falmagne and J.-P. Doignon. *Learning Spaces.* Springer-Verlag, Heidelberg, 2011.

[16] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[17] C. A. Field and A. H. Welsh. Bootstrapping clustered data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(3):369–390, 2007.

[18] C. Hockemeyer, T. Held, and D. Albert. RATH-A relational adaptive tutoring hypertext WWW-environment based on knowledge space theory. In *Proceedings of 4th International Conferenceon Computer Aided Learning and Instruction in Science and Engineering*, pages 417–423. Citeseer, 1997.

[19] T. Käser, S. Klingler, A. G. Schwing, and M. Gross. Beyond knowledge tracing: Modeling skill topologies with bayesian networks. In *International conference on intelligent tutoring systems*, pages 188–198. Springer, 2014.

[20] D. Lynch and C. P. Howlin. Real world usage of an adaptive testing algorithm to uncover latent knowledge. In *Proceedings of the 7th International Conference of Education, Research and Innovation*, pages 504–511, 2014.

[21] J. Matayoshi and E. Cosyn. Identifying student learning patterns with semi-supervised machine learning models. In *Proceedings of the 26th International Conference on Computers in Education*, pages 11–20, 2018.

[22] J. Matayoshi, E. Cosyn, and H. Uzun. Are we there yet? Evaluating the effectiveness of a recurrent neural network-based stopping algorithm for an adaptive assessment. *International Journal of Artificial Intelligence in Education*, 31(2):304–336, 2021.

[23] J. Matayoshi, U. Granziol, C. Doble, H. Uzun, and E. Cosyn. Forgetting curves and testing effect in an adaptive learning and assessment system. In *Proceedings of the 11th International Conference on Educational Data Mining*, pages 607–612, 2018.

[24] J. Matayoshi, H. Uzun, and E. Cosyn. Deep (un)learning: Using neural networks to model retention and forgetting in an adaptive learning system. In *International Conference on Artificial Intelligence in Education*, pages 258–269. Springer, 2019.

[25] B. W. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure*, 405(2):442–451, 1975.

[26] D. M. Powers. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2:37–63, 2011.

[27] S. W. Raudenbush. Statistical analysis and optimal design for cluster randomized trials. *Psychological Methods*, 2(2):173–185, 1997.

[28] A. Reddy and M. Harper. Mathematics placement at the University of Illinois. *PRIMUS*, 23(8):683–702, 2013.

[29] T. A. Snijders and R. J. Bosker. *Multilevel Analysis: An Introduction to Basic and Advanced Multilevel Modeling.* Sage, 2011.