# Instruction-Embedded Assessment for Reading Ability in Adaptive Mathematics Software

HUSNI ALMOUBAYYED, Carnegie Learning, USA
STEPHEN E. FANCSALI, Carnegie Learning, USA
STEVE RITTER, Carnegie Learning, USA

Adaptive educational software is likely to better support broader and more diverse sets of learners by considering more comprehensive views (or models) of such learners. For example, recent work proposed making inferences about "non-math" factors like reading comprehension while students used adaptive software for mathematics to better support and adapt to learners. We build on this proposed approach to more comprehensive learning modeling by providing an empirical basis for making inferences about students' reading ability from their performance on activities in adaptive software for mathematics. We lay out an approach to predicting middle school students' reading ability using their performance on activities within Carnegie Learning's MATHia, a widely used intelligent tutoring system for mathematics. We focus on how performance in an early, introductory activity as an especially powerful place to consider instruction-embedded assessment of non-math factors like reading comprehension to guide adaptation based on factors like reading ability. We close by discussing opportunities to extend this work by focusing on particular knowledge components or skills tracked by MATHia that may provide important "levers" for driving adaptation based on students' reading ability while they learn and practice mathematics.

Additional Key Words and Phrases: intelligent tutoring systems, predictive modeling, assessments, machine learning

## 1 INTRODUCTION

Recent research [22] argues that, with more comprehensive "views" or statistical models of learning, "educational technology could better address the learning needs of the whole student...". Advanced learning technologies for mathematics instruction that address the whole student may, for example, consider "non-math" factors that are likely to impact student learning. Addressing "non-math" factors might involve addressing student motivation, affect, meta-cognition, or cognitive factors less related to math than other domains, among a variety of factors known to be related to learning. Richey et al. [22] develop a particular example, modeling reading comprehension as a "non-math" factor that may affect learning for students using Carnegie Learning's MATHia [23] intelligent tutoring system (ITS), a widely used adaptive software for middle school and high school mathematics instruction. They posit that modeling one or more "non-math" factors like

Authors' addresses: Husni Almoubayyed, halmoubayyed@carnegielearning.com, Carnegie Learning, 501 Grant St, Ste 1075, Pittsburgh, PA, USA, 15219; Stephen E. Fancsali, sfancsali@carnegielearning.com, Carnegie Learning, 501 Grant St, Ste 1075, Pittsburgh, PA, USA, 15219; Steve Ritter, sritter@carnegielearning.com, Carnegie Learning, 501 Grant St, Ste 1075, Pittsburgh, PA, USA, 15219.

reading comprehension within adaptive learning software for mathematics will enable greater personalization or individualized learning and consequently improve learning for more students. We build on this prior work, establishing empirically the usefulness of certain performance measures of student work in MATHia activities (or "workspaces") as potential embedded assessments of reading ability. We additionally focus on performance measures in an introductory activity within MATHia that Richey et al. proposed could serve as early indicators of reading ability. We present how different pieces of MATHia content may load more on reading than others, providing an empirical foundation for this choice of introductory MATHia activity. We propose new performance indicators for reading ability, describing predictive models of students' scores on standardized, end-of-year English Language Arts (ELA) exam scores. We consider reading prediction models that rely on a broad scope of activities designed to introduce math topics and develop conceptual understanding to students, as well as models that rely just on performance within the introductory MATHia activity. Finally, we lay out some opportunities for future adaptive supports based on inferred reading ability estimates within adaptive instructional systems for mathematics like MATHia.

## 2 MATHIA

MATHia (formerly known as Cognitive Tutor [23]) is an ITS for middle and high school mathematics used by hundreds of thousands of learners every year across the United States. Content in MATHia is divided into sets of multi-step problems called "workspaces", through which, students make progress as they work on prescribed sequences of content. Typical sequences of content (e.g., for Grade 7 Mathematics) are comprised of 90-120 workspaces, with variations due to factors including different state standards, teachers' and administrators' preferences, and other needs for customization. MATHia's workspaces broadly fit into two classes: "Concept Builder" workspaces (or "Concept Builders") and "Mastery" workspaces. Mastery workspaces are so-named due to their implementation of mastery learning [6] [26], relying on an atomization of math topics into fine-grained knowledge components (KCs) [17] targeted for student practice. In these workspaces, the ITS presents problems to students that emphasize KCs that they have yet to master until they reach mastery of all KCs in the workspace. Upon reaching mastery of all KCs (or working on a pre-set maximum number of problems) in a workspace, students are allowed to move on to subsequent workspaces. Student KC mastery within such workspaces is tracked using Bayesian Knowledge Tracing (BKT) [2]. Additionally, MATHia provides context-sensitive hints to students and immediate feedback on incorrect responses at individual steps within problems. A specific class of feedback, just-in-time (JIT) feedback, is especially sensitive to so-called "known misconceptions."

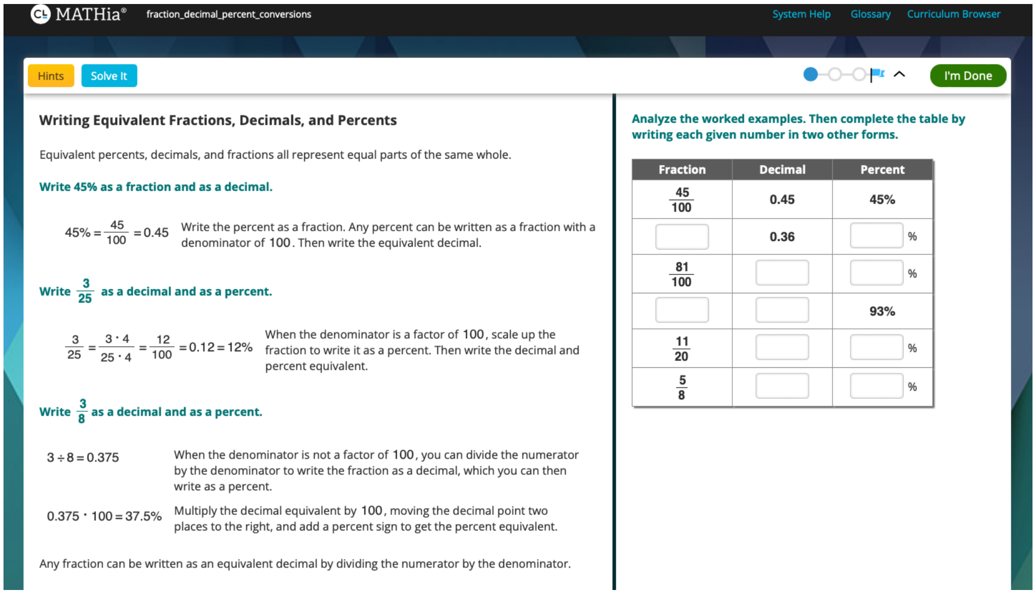
Additional support available to students in MATHia includes a glossary, worked-examples, and step-by-step examples of problems. Data collected from such activities in MATHia (and previously Cognitive Tutor) have been the subject of extensive research in learning analytics and allied research areas over the past 20+ years.

Math content that students practice in Mastery workspaces is often introduced to students in Concept Builders. With the notable exception of [22], the present work begins to address the relative dearth of research on student performance in Concept Builders, which we now briefly describe in more detail.

### 2.1 Concept Builders

Concept Builder workspaces share many of the features of MATHia's Mastery workspaces, but there are several important differences between the two types of workspaces. Like Mastery workspaces, Concept Builders present students with multi-step problems and provide students with the ability to request hints as well as provide immediate, context-sensitive JIT feedback to incorrect answers. Concept Builder problems are often accompanied by content like explanatory text, videos, or interactive tools that allow students to explore particular concepts. However, Concept Builders differ from Mastery workspaces in that they present a fixed sequence of problems to students and do not track progress to mastery of a set of KCs. Rather, students make progress through a Concept Builder by correctly answering questions and performing tasks within the workspace, generally by using information from explanatory text or videos or from the results of actions taken in interactive tools. The MATHia screenshot in Fig. 1 shows a problem from the Concept Builder "Fraction, Decimal, Percent Conversions." On the left, we see several worked-examples and explanatory text. On the right, the student is tasked with filling the empty cells in the table with appropriate values.

Later, we'll be considering the extent to which student performance in Concept Builders is related to end-of-year math performance compared to reading performance. We'll see that this particular example (i.e., "Fraction, Decimal, Percent Conversions") is one in which performance is least related to end-of-year reading performance, compared to math performance. Student performance on other Concept Builders appear to be more related to reading than mathematics. Especially if such Concept Builders appear relatively early in students' MATHia curriculum sequence, we might use performance indicators from such Concept Builders to make inferences about, build models of, and present adaptive experiences to students based on their reading ability. Richey et al. [22] treat one performance indicator from a particular Concept Builder as an indicator of reading



**Writing Equivalent Fractions, Decimals, and Percents**

Equivalent percents, decimals, and fractions all represent equal parts of the same whole.

**Write 45% as a fraction and as a decimal.**

$45\% = \frac{45}{100} = 0.45$ Write the percent as a fraction. Any percent can be written as a fraction with a denominator of 100. Then write the equivalent decimal.

**Write $\frac{3}{25}$ as a decimal and as a percent.**

$\frac{3}{25} = \frac{3 \cdot 4}{25 \cdot 4} = \frac{12}{100} = 0.12 = 12\%$ When the denominator is a factor of 100, scale up the fraction to write it as a percent. Then write the decimal and percent equivalent.

**Write $\frac{3}{8}$ as a decimal and as a percent.**

$3 \div 8 = 0.375$ When the denominator is not a factor of 100, you can divide the numerator by the denominator to write the fraction as a decimal, which you can then write as a percent.

$0.375 \cdot 100 = 37.5\%$ Multiply the decimal equivalent by 100, moving the decimal point two places to the right, and add a percent sign to get the percent equivalent.

Any fraction can be written as an equivalent decimal by dividing the numerator by the denominator.

**Analyze the worked examples. Then complete the table by writing each given number in two other forms.**

| Fraction | Decimal | Percent |
|---|---|---|
| $\frac{45}{100}$ | 0.45 | 45% |
| | 0.36 | % |
| $\frac{81}{100}$ | | % |
| | | 93% |
| $\frac{11}{20}$ | | % |
| $\frac{5}{8}$ | | % |

Fig. 1. Screenshot of the first problem in MATHia's "Fractions, Decimals, Percent Conversions" Concept Builder.

comprehension ability, MATHia's introductory "Pre-Launch Protocol" Concept Builder. We now consider this Concept Builder in more detail.

## 2.2 The "Pre-Launch Protocol" Concept Builder Workspace

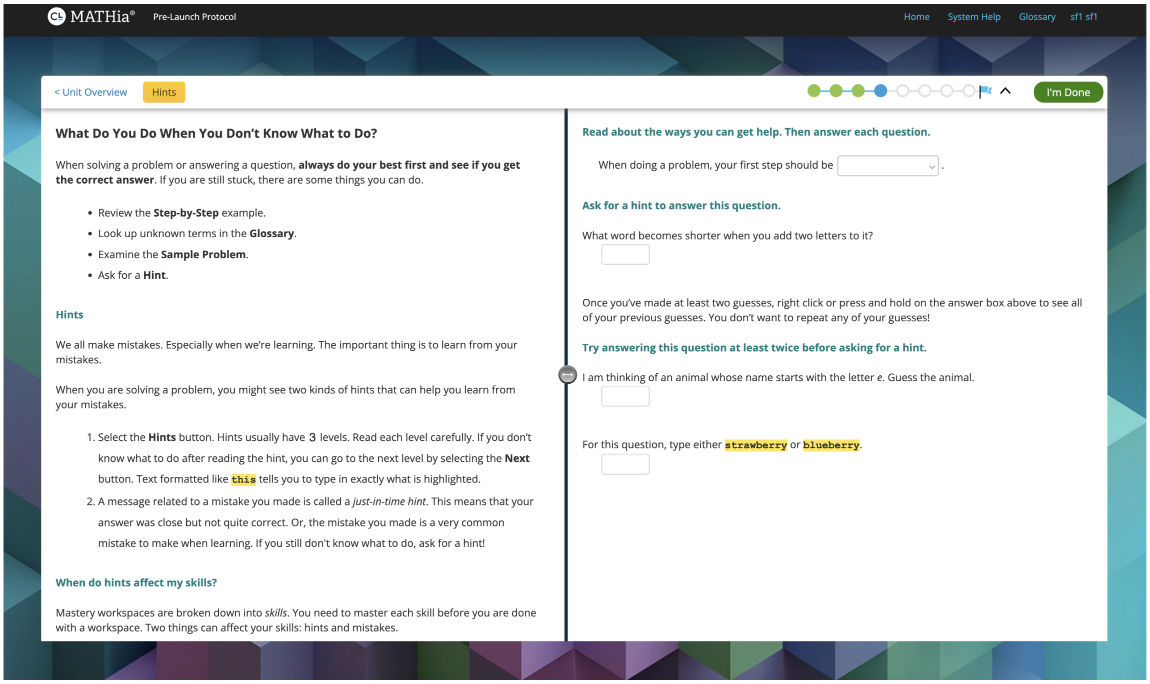
The first workspace that nearly all learners using MATHia encounter is an introductory activity presented as a Concept Builder called "Pre-Launch Protocol." This Concept Builder is intended to introduce various learning tools (e.g., a glossary), user-interface features (e.g., an expression and equation solver), and other supports for learning (e.g., context-sensitive hints) that learners will encounter and be able to use as they work through MATHia workspaces. The "Pre-Launch Protocol" Concept Builder is unique for at least two reasons: First, as the first piece of content that students encounter, being able to use it to learn more about students' reading ability can be particularly powerful early in the school year. Second, "Pre-Launch Protocol" is not designed to require any grade-level math knowledge from students, and therefore, performance on it is likely linked to non-math factors like reading.

The screenshot in Fig. 2 shows the fourth problem within "Pre-Launch Protocol." Steps within this problem are designed to compel students to engage with the hint features of MATHia by, for example, asking students to answer riddles and similar questions (e.g., to determine about which animal name beginning with the letter "e" that the tutor is "thinking"). The problem is designed to make sure that students understand that hints are available to them and to understand the UI elements associated with asking for a hint.

Richey et al. [22] use student performance in “Pre-Launch Protocol” as an indicator of reading comprehension ability, *prima facie* assuming that the Concept Builder functions more like a test of reading comprehension than as an assessment of math skills. With access to data about student performance on end-of-year ELA tests, we provide empirical support for this choice of content within MATHia as a place to seek data-driven performance indicators that might serve as embedded diagnostic assessments of reading ability to drive a more comprehensive view or model of the learner as they engage with math content. The use of “Pre-Launch Protocol” is especially powerful for this purpose. Since this Concept Builder is the first workspace that nearly all students encounter, diagnostic inferences made based on students’ work in the workspace can potentially be used for adaptive support for the rest of their subsequent work in MATHia. We will also consider the quality of predictions or inferences made about reading ability using only “Pre-Launch Protocol” compared to using data for all Concept Builders in which students work across the entire school year. While predictions made using only the “Pre-Launch Protocol” may be more appropriate for providing reading supports to students within the same year, predictions made using all Concept Builders may be used to study historical patterns in student behavior based on reading ability, and inform content and platform improvements.

### 2.3 Predicting Summative Math Assessments using MATHia Performance Data

Prior work on predicting *mathematics* end-of-year standardized test scores from math ITS data has been highly successful and scalable. Ritter et al. [24] found that MATHia data (at the time known as Cognitive Tutor) is predictive of student performance on end-of-year math tests. Using data for three thousand middle-school students, Ritter et al. [24] were able to model student performance, improving predictions over using pre-tests alone. At a larger scale, Zheng et al. [31] used MATHia data in addition to student demographic and prior knowledge data to predict end-of-year math standardized test achievement levels for tens of thousands of students in the Miami-Dade County Public Schools at the middle school level, achieving an average off-by-one error rate below 1. In a different study, Pardos et al. [20] used so-called “detector” models of behavior and affective states [4], including detectors of gaming the system [3], boredom, engaged concentration, confusion, and frustration. Using ASSISTments [12] data on more than one thousand 8th grade students over 2 years, Pardos et al. [20] found that many of these behaviors correlate significantly with high-stakes math exams.



**What Do You Do When You Don’t Know What to Do?**

When solving a problem or answering a question, **always do your best first and see if you get the correct answer**. If you are still stuck, there are some things you can do.

- Review the **Step-by-Step** example.
- Look up unknown terms in the **Glossary**.
- Examine the **Sample Problem**.
- Ask for a **Hint**.

**Hints**

We all make mistakes. Especially when we’re learning. The important thing is to learn from your mistakes.

When you are solving a problem, you might see two kinds of hints that can help you learn from your mistakes.

1. Select the **Hints** button. Hints usually have 3 levels. Read each level carefully. If you don’t know what to do after reading the hint, you can go to the next level by selecting the **Next** button. Text formatted like **`this`** tells you to type in exactly what is highlighted.
2. A message related to a mistake you made is called a *just-in-time hint*. This means that your answer was close but not quite correct. Or, the mistake you made is a very common mistake to make when learning. If you still don't know what to do, ask for a hint!

**When do hints affect my skills?**

Mastery workspaces are broken down into *skills*. You need to master each skill before you are done with a workspace. Two things can affect your skills: hints and mistakes.

**Read about the ways you can get help. Then answer each question.**

When doing a problem, your first step should be ______ .

**Ask for a hint to answer this question.**

What word becomes shorter when you add two letters to it?

Once you’ve made at least two guesses, right click or press and hold on the answer box above to see all of your previous guesses. You don’t want to repeat any of your guesses!

**Try answering this question at least twice before asking for a hint.**

I am thinking of an animal whose name starts with the letter *e*. Guess the animal.

For this question, type either **`strawberry`** or **`blueberry`**.

Fig. 2. Screenshot of the fourth problem in MATHia’s “Pre-Launch Protocol” Concept Builder, which is set up in such a way that students will almost certainly need to ask for hints to provide a correct response to a prompt to identify an animal whose name begins with the letter “e.” The tutor isn’t thinking of “elephant.”

These previous studies present ambitious approaches to the potential for instruction-embedded assessments for summative outcomes like end-of-year test scores for the instructional domains (i.e., math) targeted by the systems. It is less clear that a math ITS such as MATHia, that is not designed to assess reading skills, would include enough data to allow for predicting reading ability (or end-of-year ELA test scores); however, there have been many studies that showed that math and reading comprehension are closely related (see, e.g., [11, 18, 19, 28]). The goals of the present study differ: a major goal of predicting end-of-year standardized math ability from a math ITS may be to develop high-quality progress monitoring metrics that do an exceedingly good job of predicting student outcomes on standardized tests (e.g., to provide helpful insight to instructors throughout the year about student progress toward end-of-year test success). By predicting reading ability from a math ITS, however, we do not aim to replace standardized ELA tests. Rather, we aim to use these predictions to personalize potential supports, interventions, and other facets of the learner experience for students based on non-math factors. More comprehensive learner models that use both embedded math and ELA assessments, and perhaps other data-driven approaches like affective detectors [4], are likely to drive improved learning experiences and outcomes, in more trustworthy platforms, for all students, beyond the current state-of-the-art.

## 3 DATA

We had access to student action-level data from the MATHia platform for a set of 2,685 Grade 6-8 learners from a school district within a mid-western US state for whom we were provided access to end-of-year state test scores for both mathematics and ELA for the 2020-2021 academic year. Both end-of-year state tests have five achievement levels into which scores fall. These achievement levels represent cut-off points that categorize students based on their test scale scores (which vary between 562 - 847 for ELA and 623 - 806 for math within the district we studied). In this district, levels 1 through 3 correspond roughly to students in the bottom half of scores, while levels 4-5 represent the top half. For example, 48% of students in 7th grade are in levels 1 through 3, with 52% falling in levels 4 and 5. We use these levels as our prediction targets in Section 5. Schools in this state refer to student performance in levels 1 to 5 as Limited, Basic, Proficient, Accelerated, and Advanced, respectively, with levels 3 and above treated as a passing grade.

Across all grades and Concept Builder workspaces, we use data collected on 7,817,080 actions at problem steps, as students interact with Concept Builders within MATHia. Within the Pre-Launch Protocol alone, data were collected on 164,994 step-level actions taken by these 2,685 students.

While Carnegie Learning creates specific sequences of content targeting each grade level (named Courses), some students may get assigned some material from Courses below or above their target Course. Specifically, Courses 1, 2, and 3 target grade levels 6, 7 and 8, respectively. While the majority of student action data that we have collected for 7th grade students belongs to Course 2, there are still many students who completed Course 1 and Course 3 content, and we include that data in this analysis.

In most of the analysis presented, we use data for 7th grade students as an example. Our analysis excludes workspaces that were attempted by fewer than 100 students (or approximately 11% of students in 7th grade) as well as a single workspace in MATHia Course 1 (Grade 6 Math) that showed a small but statistically significant ($r = -0.15, p < 10^{-3}$) negative correlation between both MATHia performance and ELA scores and MATHia performance and Math scores. We suspect the correlations for this outlier workspace are due to some students engaging in unproductive behavior such as gaming the system [3], but additional investigation is needed. Workspaces with smaller sample size are not uniform: typically, these workspaces are either at the very end of the curriculum or above grade level (attempted by the highest achieving students) or below the grade level (attempted by the lowest achieving students). Nevertheless, data from these workspaces were too noisy and sparse to be useful for our present goals of understanding the relationship between MATHia performance and reading ability, and finding diagnostic performance indicators for reading ability over broad populations of students.

## 4 PERFORMANCE IN MATHIA AND READING ABILITY

We start by examining the relationship between students' end-of-year or summative ELA scores with their performance on each MATHia Concept Builder, defined as the percentage of correctness on first attempts at every problem-step within a workspace. Correctness on first attempts at problem-steps is a common performance metric for data collected from ITSs like MATHia [16]. We find that the correlation between the ELA scores and this metric of MATHia performance varies across workspaces, ranging between 0.11 and 0.61 for students in Grade 6 across 84 workspaces, 0.11 and 0.47 for students in Grade 7 across 85 workspaces, and 0.14 and 0.51 for students in Grade 8 across 33 workspaces at statistically-significant levels (p-value<0.05). We also identified Concept Builders for which no statistically-significant correlation was found (63, 32, and 21 workspaces in Grades 6, 7, and 8, respectively).

We consider the possibility that the correlation between MATHia workspace performance and ELA test scores is actually due to an underlying correlation between students' math ability and reading ability (i.e., between their math and ELA test scores). We find that math ability and reading ability (as measured by these test scores) are correlated; for example, this correlation is ($r = 0.675, p < 10^{-140}$) for Grade 7 students (with similar correlations across grades). To partially mitigate issues of overall math ability (via math scores) being a confounding factor, we define a metric to identify workspaces that are the most and least correlated to ELA scores compared to their correlations to Math scores. First, for each workspace, $w$, we define the quantity

$$\rho^w = \frac{r^w_{\text{ELA}}}{r^w_{\text{Math}}}, \tag{1}$$

where $r_{\text{ELA}}$ is the correlation between MATHia performance and ELA scores, and $r_{\text{Math}}$ is the correlation between MATHia performance and Math scores. We then use the Z-score of equation 1 as our metric, i.e., the metric we use is:

$$Z_\rho = \frac{\rho^w - \bar{\rho}}{\sigma(\rho)} \tag{2}$$

Workspace-level correlations with ELA scores vary to a greater degree than correlations with Math scores, indicating that the former may contain "signal" not present in the latter ($\sigma_{\text{ELA}}/\mu_{\text{ELA}} = 1.23, 1.03, 1.76$ for Grades 6, 7, and 8 respectively, whereas $\sigma_{\text{Math}}/\mu_{\text{Math}} = 0.79, 0.75, 0.64$ for Grades 6, 7, and 8 respectively. Section 5.4 will provide the strongest evidence that the signal captured in

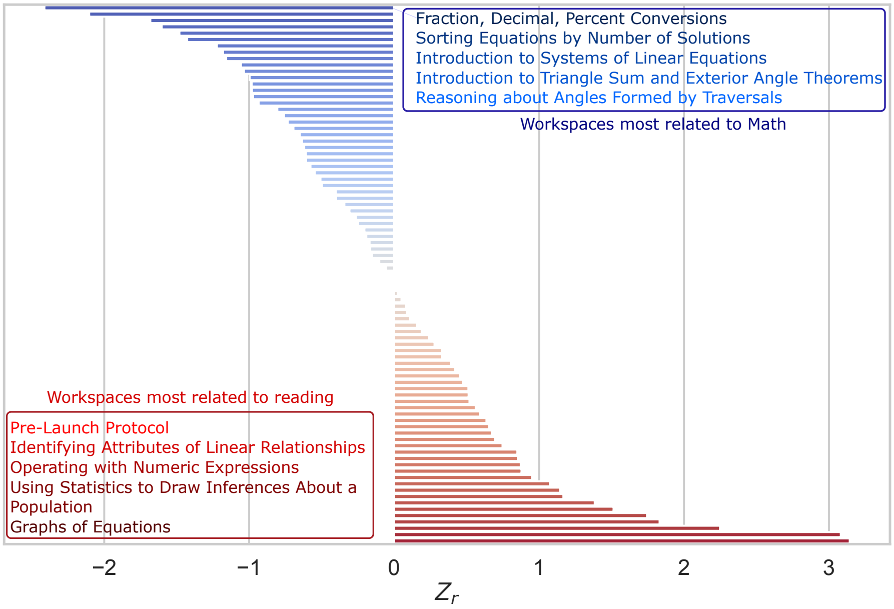


Fig. 3. A ranking of every Concept Builder workspace attempted by at least 100 Grade 7 students with a statistically significant correlation to both math and ELA test scores. The ranking is based on the $Z_\rho$ metric defined in 4. The Concept Builders that are ranked as the top 5 most and least correlated to reading, compared to their correlation with math, are also named explicitly within the plot. The Pre-Launch Protocol appears as fifth most reading-related, and is in the top 5 for all grade levels.

this analysis is not due to overall math ability (or some other more general measure of academic aptitude) being a confounding factor.

Figure 3 shows this metric for all non-excluded workspaces that Grade 7 students attempted. The five workspaces with highest and lowest metric values are identified in the Figure. “Pre-Launch Protocol” is fifth highest when ranked by this metric. For Grade 6 students, the “Pre-Launch Protocol” ranks the highest, and for Grade 8 students, it ranks third. This confirms our expectations and the judgment of the Concept Builder’s content made by Richey et al. [22]. While performance in the “Pre-Launch Protocol” can be viewed as an assessment of reading ability, especially compared to the majority of other MATHia content, there are other workspaces that seem to be similarly reading-related, with a few at even higher degrees. The workspaces with the highest and lowest $Z_\rho$ values are also in line with expectations. For example, “Graphs of Equations” includes word problems where students need to relate elements of the word problem to linear equations and graphs of these equations. In contrast, the Concept Builder “Fraction, Decimal, Percent Conversions,” (see Fig. 1) has the lowest metric value. This workspace is by no means devoid of words, but the activity on which student correctness is measured is limited to students converting one of the three forms of numbers (fractions, decimals, and percentages) to the other two – an activity that does not appear to be related to reading skills (or at least students appear to be able to carry out this task without necessarily being able to perform at high levels on their end-of-year ELA exam).

## 5 PREDICTING END-OF-YEAR READING SCORES

Given what we found in Section 4, it is natural to consider the question of whether the fact that students' MATHia performance correlates with reading scores at different levels across Concept Builders may be useful to predict end-of-year ELA test scores, similarly to how students' MATHia performance is predictive of end-of-year Math test scores. Notably, our goals in predicting end-of-year ELA and Math scores are different. In predicting reading ability within a platform that targets math learning, our goals are high-quality (precise) predictions. With these predictions, one can (a) infer as early as possible that a student may benefit from supports for reading, or (b) later on study and model learner behavior based on reading ability to develop content and platform improvements. While MATHia and its developers would similarly like to know that a student might benefit from some type of just-in-time math support, perhaps outside the system, (see, e.g., [30] [10]), building models that predict end-of-year *math* test scores can also be helpful even if they accumulate evidence of learning over the course of the entire school year, providing, for example, progress monitoring to teachers [20] [30].

We compare our following models to the predictor used by [22], the "assistance score" [16] on the Pre-Launch Protocol, as a baseline predictor. The assistance score is defined as the sum of the count of hints requested and the count of errors made by a student. Using this assistance score to predict below median students (nearly equivalent to levels 1-3) has a FPR of 32% when tested on ELA levels; and when tested against Math scores, it achieves very similar precision, with a FPR of 35%, making it unclear how much purely "reading signal" is captured by the assistance score. We find that the correlation between the assistance score and ELA test scores is -0.56, while the correlation between the assistance score and Math test scores is -0.51, both with p-values below $10^{-10}$.

### 5.1 Feature Engineering

Our first step in building predictive models of ELA exam scores was to consider additional features that might capture important information about student performance in Concept Builder workspaces. We identify multiple features, that we define at the step level for each student, to be used in addition to correctness on first attempt. These features are the total number of hints requested, the total number of just-in-time (JIT) feedback prompts received (corresponding to errors that correspond to “known misconceptions” as tracked by MATHia), and the total number of attempts at each step. When averaged over every problem-step (over all Concept Builder workspaces) for each student, all of these features correlate at a statistically-significant level with ELA scores. Respectively, the correlation between ELA scores and each of the average number of attempts, hints, and JITs is -0.29, -0.4, and -0.23, respectively, all with p-values below $10^{-10}$.

### 5.2 Predictive Models Using All Concept Builders

We frame our task as one of classifying student ELA achievement levels, rather than a regression task of student ELA scale scores. This is due to the fact that our goal is identifying struggling readers, and we take distinguishing differences between student scores intra-level to be unnecessary to achieve this goal. For example, we don’t immediately envision driving interventions or adaptive reading supports that would be sensitive to small differences within the same achievement level. Practically, training a predictive model on finer-grained scale scores, which could include more noise, may lead the models to be more likely to over-fit. Further, we take the higher-level classification (e.g., that a student is at or above a particular level of “proficiency”) to generally be of more practical importance for stakeholders like instructors and school administrators.

Given that students have different reading ability levels in different grades, and typically work through different sets of Concept Builders, we do not combine data across student grades in our predictive models. We present results for students in Grade 7, chosen due to the fact that it has the largest number of workspaces included by our criteria. Results from different grade levels were broadly similar to findings for Grade 7.

We start by using the all Concept Builder data to train a machine learning model to correctly predict the end of year ELA levels. We use the following metrics to evaluate and compare models:

- *Confusion Matrices* show the fraction of true levels (i.e., actual ELA exam achievement levels) that fall in each predicted class. We present confusion matrices normalized by predicted class (such that the values of each column in the matrix sum up to 1), so as to not de-emphasize classes with lower frequencies. Normalizing by the predicted class, rather than the true class, is equivalent to the question of ‘given our predictions, how much of each true class falls within it?’ This question is the most suitable given our goal of identifying a sample of students who are predicted to be struggling readers, to make inferences about adaptive supports they may need for reading, with minimum contamination of higher performers who may be potentially de-motivated by such supports.

- *Mean Absolute Error (MAE)* is defined as the absolute-value difference between the predicted classes and the true classes, averaged over predictions.
- *Below median False Positive Rate* (FPR) is defined as the percentage of students with ELA achievement levels of 4-5 (approximately above median) that are incorrectly predicted to be in ELA achievement levels 1-3. We are primarily concerned in this work with identifying students who are struggling readers to better support such learners, so this metric helps us better understand the extent to which our modeling efforts may help us achieve that goal.

We compare the performance of three machine learning models:

- Neural Network (multi-layer perceptron; MLP. See e.g., [5]): with a 100-node single hidden layer and rectified linear unit function activation. We decided on this simple architecture after observing that a multi-hidden layer network performed worse due to over-fitting, and a multi-layer network regularized by Dropout [27] performed as well as the un-regularized single layer network. We use a categorical cross-entropy loss function and a stochastic gradient-based optimizer by [15] with an adaptive (decreasing) learning rate.
- Support Vector Machine (SVM) classifier (see, e.g., [9]): with a one-versus-rest scheme and a radial basis function kernel. The $L_2$ regularization parameter C was chosen using 2-fold cross-validation (C=3).
- XGBoost: Gradient boosted decision tree classifier [7], with a 100 maximum tree-based learners, a depth-wise growth policy and $L_2$ regularization.

To train the classifiers, we categorized the problem-step-level features we engineered into four categories:

- `correct`: Whether the student's first attempt at this step is correct (1) or incorrect (0).
- `attempt`: The total number of attempts the student has made on this problem step (can be 0 or higher).
- `jit`: The number of JIT feedback prompts the student has received at this problem step (can be 0 or higher).
- `hint`: The number of hints the student requested at this problem step (between 0 and 3 for the majority of steps, but may be up to 4 in rare cases).

Within each category of features, we use each problem step attempted by more than 100 students as a feature. [1] We train the models on each of these categories separately, and then use the mode of the predictions of the models as the final "majority ensemble" classifier. We find that this yields significantly better performance than training the model on all the features (across categories) at once.

[1] We find that the performance degrades when using either a stricter or looser requirement.

| Classifier | correct | attempt | jit | hint | Ensemble |
|---|---|---|---|---|---|
| SVM | 0.740 | 0.761 | 0.813 | 0.816 | 0.740 |
| XGBoost | 0.805 | 0.870 | 0.883 | 0.820 | 0.785 |
| MLP | 0.796 | 0.787 | 0.824 | 0.876 | 0.746 |

Table 1. A comparison of the mean absolute error (MAE) of three machine learning models trained separately on four categories of features, and an ensemble predictor that takes the majority vote of the four classifiers.

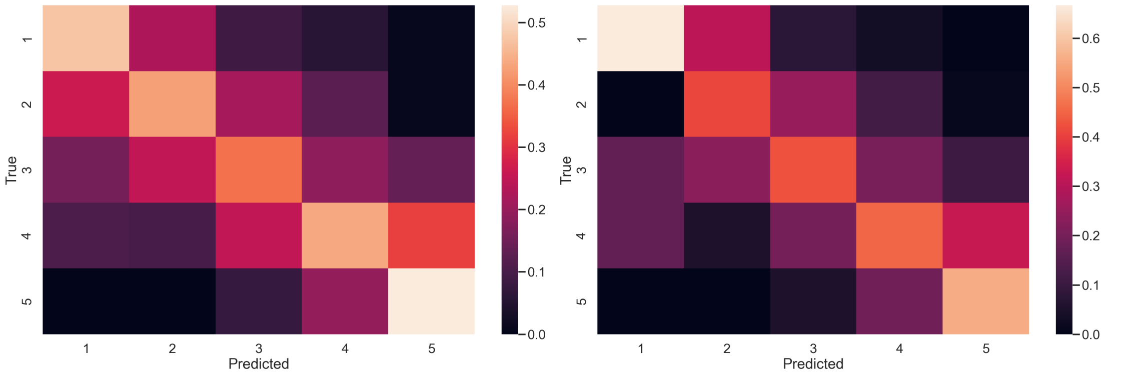


Fig. 4. Confusion matrices for predictions of student ELA levels against true ELA levels, using an ensemble majority classifier (left), and an ensemble ‘strong’ majority classifier (right). The matrices are normalized such that every column sums up to 1. Note the difference in color scales. The confusion matrices show that the models are correctly identifying the majority of students in each level, and misclassifications are typically small, mostly within an adjacent class.

The data is split into two equal halves, each containing half the students at random. One half is reserved for training the models, and the other is reserved for testing the models. Model performance in what follows is reported as measured entirely on the test set.

Table 1 shows the MAE of each of the classifiers on each category, as well as the majority ensemble. While the SVM classifier has slightly lower mean absolute error than the MLP Classifier, upon examining its confusion matrix, we find that it fails completely at making any level 1 predictions, and thus treat the ensemble MLP classifier (which does not have this problem) as the best classifier. Both the MLP and XGBoost classifiers improve significantly when using the majority ensemble classifier.

In addition to the MAE, we look at the FPR, i.e., the percentage of approximately above median readers (students whose actual performance falls within level 4 and 5) that are incorrectly classified as approximately below median (falling within levels 1-3). This percentage was 21% for the ensemble MLP classifier.

*5.2.1 Obtaining a smaller, more precise sample.* In some cases, such as identifying struggling readers from historical data to fit BKT parameters for them more appropriately, it is important to

| Classifier | Baseline | Standard majority | Strong majority | X3 |
|---|---|---|---|---|
| FPR | 32% | 21% | 7% | 10% |
| Retained Predictions | 100% | 100% | 60% | 55% |

Table 2. The FPR for the assistance score basline, the standard majority classifier, as well as its modified versions, the strong majority and X3 classifiers, when trained on the entire Concept Builder data

be able to obtain a predicted sample with lower FPR than 21%. This is equivalent to improving the precision at the expense of lower recall. We compare two methods to do so:

- *'Strong' Majority* classifier: in which case, we only take into account predictions where 3 or 4 of the (`correct`, `hint`, `jit` and `attempt`) classifiers agree.
- *X3* classifier where we use predictions of students falling within levels 1-2 as being below median, rather than in levels 1-3.

Both of these methods lead to improved FPR for fewer predictions. In particular, only predictions that have a majority of 3 or 4 predictions will be 'retained' for the Strong majority classifier, and only the predictions that fall into levels 1 and 2 will be retained for the X3 classifier.

We find that 60% of predictions have a 'strong' majority, and using those alone, we achieve an MAE of 0.66 and an FLR of 7%, while the X3 classier retains 55% of the predictions achieving an FPR of 10%, making the 'Strong' majority classifier superior in that it retains more predictions and achieves a lower FPR. Table 2 summarizes the FPR of the standard ensemble, the strong majority, and the X3 classifier, and compares their performance to the performance of the assistance score baseline. Figure 4 shows the confusion matrices for both the standard majority and strong majority classifiers, both showing consistently high quality predictions across achievement levels, with the majority of mis-classifications falling within an adjacent achievement level.

### 5.3 Predictive Models Using the Pre-Launch Protocol

Given the uniqueness of "Pre-Launch Protocol" as the first workspace nearly all students work through each year, we focus on whether student performance on this Concept Builder alone is enough to predict students' reading abilities to some degree. We use the same MLP classifier, but train it on "Pre-Launch Protocol" data only, again using data from Grade 7 students as an example, and again training on step-level data for the `correct`, `attempt`, `hints`, and `jit` categories separately, before using a majority ensemble predictor. This model achieves a MAE of 0.99, 0.98, 0.98, 0.99 for the `correct`, `attempt`, `hints`, and `jit` categories, respectively, while the ensemble majority predictor achieves a MAE of 0.90. This means that even just using the Pre-Launch Protocol, on average, the true level falls less than one achievement level away from the predicted level. The FPR for the standard majority classifier is 30%, while the FPR for the X3 classifier, which retains

| Classifier | Baseline | Standard majority | Strong majority | X3 |
|---|---|---|---|---|
| FPR | 32% | 30% | 21% | 9% |
| Retained Predictions | 100% | 100% | 36% | 43% |

Table 3. The FPR for the assistance score baseline, the standard majority classifier, as well as its modified versions, the strong majority and X3 classifiers, when trained only on the Pre-Launch Protocol

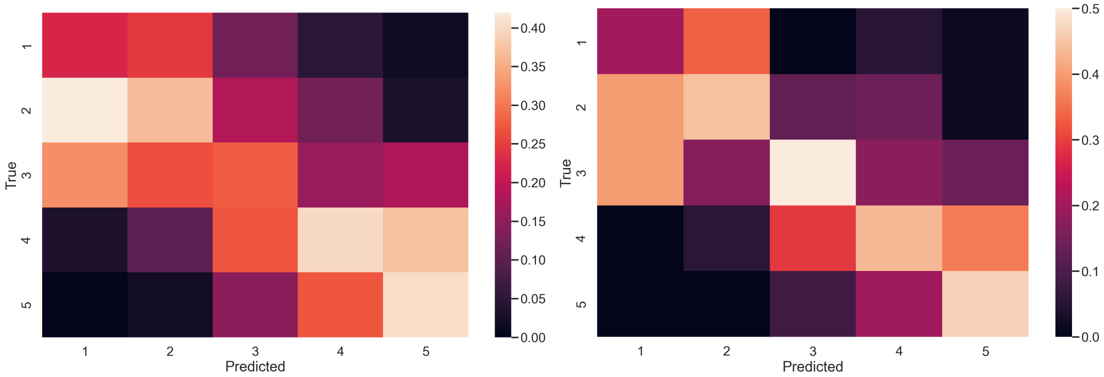


Fig. 5. Confusion matrices of predicted against true ELA levels, using an ensemble majority classifier (left) and an ensemble 'strong' majority classifier (right), trained only on Concept Builder data. While these models perform worse than the models trained on all Concept Builder data, they use significantly less data and can be used very close to the start of the academic year. The confusion matrices are normalized such that every column sums up to 1. Note that the color-scale range is higher for the 'strong' majority classifier. For the majority of predictions, the true level falls in or within an adjacent level.

36% of the data was a much lower 9%, and the FPR for the 'strong' majority classifier, which retains 43% of the data was 21%. Table 3 summarizes these FPR values and compares them to the assistance score baseline. Figure 5 shows the confusion matrices for the standard majority and strong majority classifiers, showing higher mis-classification rates than when using all Concept Builder data, but the majority of mis-classifications remain in an adjacent achievement level. Tables 2 and 3 show that all the classifiers discussed in this work perform better than the baseline assistance score classifier.

### 5.4 Comparing Prediction Outcomes to Math Scores

One major concern is whether our predictive models may be picking up on math ability or a related confounding factor, rather than on reading ability. This concern is justified, as students' ELA and Math test scores are correlated as we noted in Section 4. Figure 6 shows the ELA and Math scores for the 7th grade students that were included in this analysis, where the significant correlation between Math and ELA test scores is readily apparent. Our goal is to correctly identify students below some ELA level (e.g., below the horizontal line in Figure 6), and not below some Math level.

While rigorously establishing causal links is beyond the scope of this work, we look to provide evidence against math being a confounding factor in the relationship between MATHia performance and reading scores. Specifically, we look at whether the MLP classifier, when trained to predict ELA levels, is additionally, incidentally predictive of Math scores. If it were, then Math ability (or some more general unmeasured factor) would be a likely confounding factor. Fortunately, we find that this classifier, when trained to predict ELA levels, is a very poor predictor of Math levels. Figure 7 shows the confusion matrices of predictions of student ELA levels compared to true Math levels. The matrices are normalized by predictions (columns). Unlike the confusion matrices in Figure 4 which had the highest values across the diagonal (indicating correct predictions), the confusion matrices in Figure 7 do not; they do not frequently make correct predictions of math test scores. For each prediction, no more than 35% of true math test scores falls within *any* level.[2].

The fact that the ELA predictor do very well on predicting ELA test scores but very poorly when predicting Math test scores is reassuring that the predictor is picking up reading ability "signal," and not some underlying math signal.

## 6 CONCLUSION AND FUTURE WORK

We build on recent research [22] calling for the development of more comprehensive models of learners within adaptive learning software. Building on the specific "non-math" factor, reading ability, and adaptive learning software, MATHia, explored in this recent work, we explore empirically how reading is correlated with learning mathematics content in the MATHia ITS. Using students' end-of-year ELA test scores, in addition to their mathematics scores, we were able to find that student performance in different MATHia content was correlated to ELA scores to varying degrees. In particular, we focused on one piece of content, the "Pre-Launch Protocol," due to its high correlation to reading, and its unique position as the very first piece of content that students interact with within MATHia. The predictive power of performance in content placed early in the MATHia user experience presents important opportunities to use early student performance as an embedded diagnostic assessment to adaptively determine whether students may need additional support for reading as they use MATHia throughout the rest of the school year.

Our modeling approach relied on a simple neural network architecture to build and train a model to predict students' ELA test scores from their performance in MATHia content, particularly, their first attempt correctness, the number of hints they request, the number of just-in-time feedback prompts they receive, and the number of attempts they make on each problem-step. Our model achieved high precision, for example, out of the students that were predicted to be approximately

[2] Note the much narrower range of color-scales within the Figure compared to previous confusion matrices

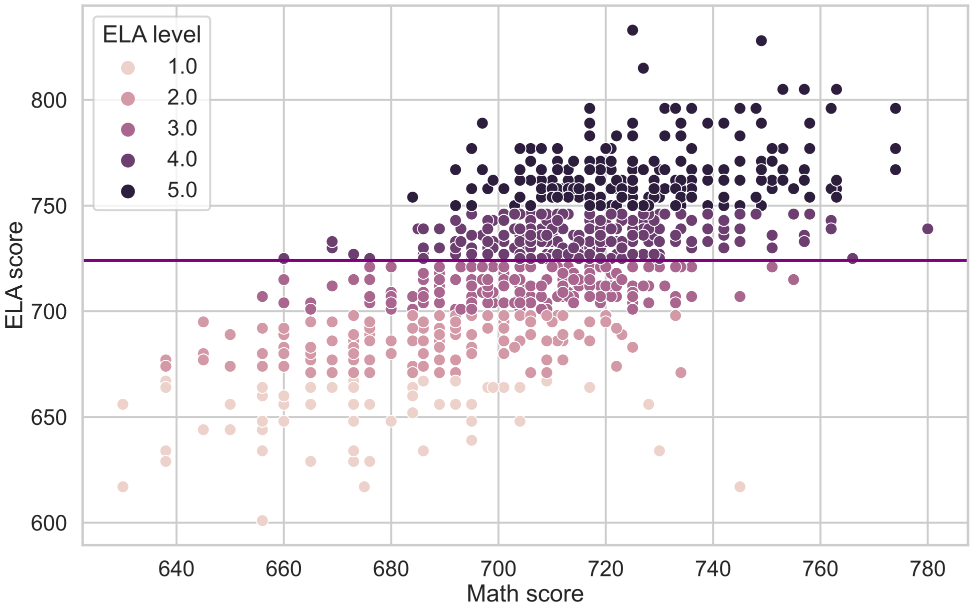


Fig. 6. The 7th grade students' ELA and Math end-of-year test scores. Students with ELA score under 725 are in levels 1 through 3, and are considered approximately below median performers (the median being 727). A horizontal line at 725 is also presented for reference. These scores are clearly correlated, making the problem of predicting ELA ability, without Math ability being a confounding factor, difficult. Fig 7, however, shows that although our classifiers work very well at predicting ELA test scores, they are extremely poor when tested against Math scores, meaning the signal being captured by the classifiers is due to reading ability, and not Math ability.

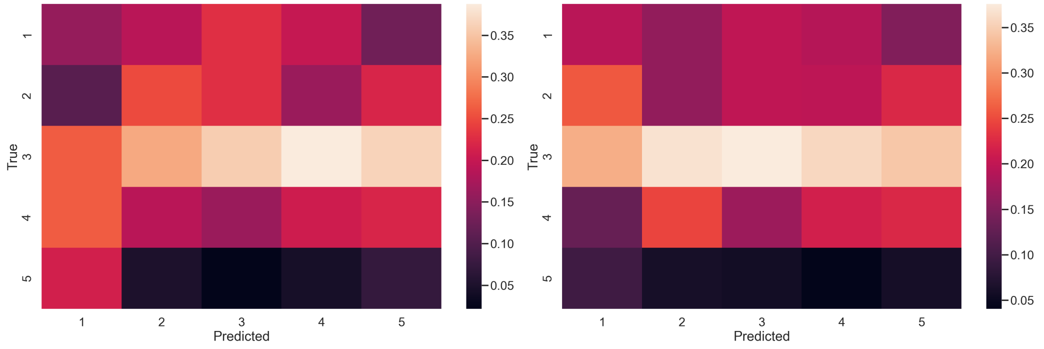


Fig. 7. Confusion matrices for predictions of student ELA levels against true *Math* levels, for the ensemble majority Concept Builder model (left), and the ensemble majority Pre-Launch Protocol model (right). We use these matrices to confirm whether the classifiers are being influenced by a math ability as a confounding factor; however, it is clear that the these predictions are extremely poor when compared to *Math* levels, which confirms that our classifiers are correctly picking up on reading ability signal, and are not being influenced by math ability as a confounding factor.

below median readers, 79% were correct predictions. We were able to improve that number to 90% for more certain predictions, which might be useful in cases where a higher precision sample is needed. We trained a similar model on the early "Pre-Launch Protocol" workspace separately, representing only one of the 85 Concept Builders included in the analysis, and achieved 70% true positive rate when predicting approximately below median readers, and were able to similarly improve that number beyond 90% in some cases.

The ability to predict reading ability from MATHia performance enables a wide variety of future research and ITS feature development. A model trained on all Concept Builder data may be used on historical data to identify how struggling readers used and performed in MATHia differently (or perhaps not differently) compared to other students, without necessarily having to rely on data from assessments provided by third parties like school districts. A model only trained on "Pre-Launch Protocol" data can be used to identify struggling readers early on in the year, which would allow for more suitable, potentially reading-specific, interventions to be shown to students when they struggle with a piece of content, particularly if that piece of content comes from a workspace that is highly-correlated with reading ability. Additionally, it may be possible to build a more flexible model that can be used to make better predictions at regular intervals, say, after collecting additional MATHia usage data each month.

In this work, we only considered one type of MATHia content, Concept Builders, for predictive modeling. In addition to Concept Builders, MATHia includes "Mastery" workspaces, that uses a BKT-estimated student model to assess student mastery of knowledge components. This type of content has the potential to be used for predictive models similar to those we consider in the present work. However, "Mastery" workspaces are also inherently different from Concept Builders in that, unlike Concept Builders, students generally solve different problems than their peers (a typical MATHia Mastery workspace includes 200 problems), which means they cannot be used for predictions at the 'step' level as we have done here. Nevertheless, there are a variety of ways that we might fruitfully aggregate or otherwise treat student work in mastery content differently (e.g., by considering correlations of performance on particular KCs to reading ability and ELA scores). We leave these approaches for future work, noting that aggregating performance in Concept Builders always led to worse performing models, so the inclusion of aggregate performance on Mastery workspaces may not yield better prediction performance.

As more data on students' end-of-year ELA scores, socio-demographics, and other possible "non-math" factors are made available to us, we aim to improve these models, for example, by examining whether training them on different student populations (e.g., different student end-of-year math levels or demographic categories) may yield different results, furthering work toward the goal of more comprehensive views and models of learners, and building a higher level of trustworthiness into intelligent tutoring systems. Confirming that the predictions from these models are not biased towards specific groups, whether demographic, geographical, or otherwise, is an essential next step to ensure fairness - a step that will become possible with these additional data.

Advances in digital experimentation on learning platforms, such as [25], also enable experimental approaches that can take advantage of early predictions of factors like reading ability to better understand if interventions are having a causal impact on particular sub-populations of learners. One might, for example, combine modeling of Concept Builder data with MATHia's Mastery workspace KC models to personalize students' knowledge tracing models (e.g., setting statistical parameters to give some students more practice on particular KCs that might be related to reading) based on reading ability inferred near the beginning of the school year. Over time, one could also re-assess whether students appear to need this sort of additional practice or some other adaptive support, as relatively low-stakes and unintrustive reading assessments might be embedded throughout MATHia Concept Builders.

Many of these envisioned approaches and experiments would have been impossible in the past without relying on relatively "intrusive" approaches to assessment (e.g., giving students some sort of pre-assessment of reading skills, likely sacrificing valuable instructional time). We see these results as promising steps toward more comprehensive adaptation, personalized learning content, and consequently more trustworthy adaptive instructional systems.

## REFERENCES

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. https://www.tensorflow.org/ Software available from tensorflow.org.

[2] John R. Anderson and Albert T. Corbett. 1995. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction* 4 (1995), 253–278.

[3] Ryan Shaun Baker, Albert T. Corbett, Kenneth R. Koedinger, and Angela Z. Wagner. 2004. Off-Task Behavior in the Cognitive Tutor Classroom: When Students "Game the System". In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vienna, Austria) *(CHI '04)*. Association for Computing Machinery, New York, NY, USA, 383–390. https://doi.org/10.1145/985692.985741

[4] Ryan SJD Baker, Sujith Gowda, Michael Wixon, Jessica Kalka, Angela Wagner, Aatish Salvi, Vincent Aleven, Gail Kusbit, Jaclyn Ocumpaugh, and Lisa Rossi. 2012. Sensor-free automated detection of affect in a Cognitive Tutor for Algebra. In *Educational Data Mining 2012*. International Educational Data Mining Society.

[5] Christopher Bishop. 2006. *Pattern Recognition and Machine Learning*. Springer. https://www.microsoft.com/en-us/research/publication/pattern-recognition-machine-learning/

[6] Benjamin Samuel Bloom. 1968. Learning for Mastery. Instruction and Curriculum. *Regional Education Laboratory for the Carolinas and Virginia, Topical Papers and Reprints* 1.

[7] Tianqi Chen and Carlos Guestrin. 2016. XGBoost: A Scalable Tree Boosting System. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. ACM, New York, NY, USA, 785–794. https://doi.org/10.1145/2939672.2939785

[8] François Chollet. 2015. Keras. https://github.com/fchollet/keras.

[9] Corinna Cortes and Vladimir Naumovich Vapnik. 2004. Support-Vector Networks. *Machine Learning* 20 (2004), 273–297.

[10] Stephen E. Fancsali, Kenneth Holstein, Michael Sandbothe, Steven Ritter, Bruce M. McLaren, and Vincent Aleven. 2020. Towards Practical Detection of Unproductive Struggle. In *Artificial Intelligence in Education*, Ig Ibert Bittencourt, Mutlu Cukurova, Kasia Muldner, Rose Luckin, and Eva Millán (Eds.). Springer International Publishing, Cham, 92–97.

[11] Peter Fuentes. 1998. Reading Comprehension in Mathematics. *The Clearing House* 72, 2 (1998), 81–88. http://www.jstor.org/stable/30189563

[12] Neil T. Heffernan and Cristina Heffernan. 2014. The ASSISTments Ecosystem: Building a Platform that Brings Scientists and Teachers Together for Minimally Invasive Research on Human Learning and Teaching. *International Journal of Artificial Intelligence in Education* 24 (2014), 470–497.

[13] J. D. Hunter. 2007. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering* 9, 3 (2007), 90–95. https://doi.org/10.1109/MCSE.2007.55

[14] Eric Jones, Travis Oliphant, Pearu Peterson, et al. 2001. SciPy: Open source scientific tools for Python. http://www.scipy.org/

[15] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, Article arXiv:1412.6980 (Dec. 2014), arXiv:1412.6980 pages. arXiv:1412.6980 [cs.LG]

[16] Kenneth R. Koedinger, Ryan S. Baker, Kyle Cunningham, Alida Skogsholm, Brett Leber, and John Stamper. 2010. *Handbook of Educational Data Mining.* CRC Press., Boca Raton, FL, Chapter A Data Repository for the EDM community: The PSLC DataShop.

[17] Kenneth R. Koedinger, Albert T. Corbett, and Charles Perfetti. 2012. The Knowledge-Learning-Instruction Framework: Bridging the Science-Practice Chasm to Enhance Robust Student Learning. *Cognitive Science* 36, 5 (2012), 757–798. https://doi.org/10.1111/j.1551-6709.2012.01245.x arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1551-6709.2012.01245.x

[18] Kenneth R. Koedinger and Mitchell J. Nathan. 2004. The Real Story Behind Story Problems: Effects of Representations on Quantitative Reasoning. *Journal of the Learning Sciences* 13, 2 (2004), 129–164. https://doi.org/10.1207/s15327809jls1302_1

[19] Janina Krawitz, Yu-Ping Chang, Kai-Lin Yang, and Stanislaw Schukajlow. 2022. The role of reading comprehension in mathematical modelling: improving the construction of a real-world model and interest in Germany and Taiwan. *Educational Studies in Mathematics* 109 (2022), 337–359.

[20] Zachary A. Pardos, R. Baker, Maria Ofelia San Pedro, Sujith M. Gowda, and Supreeth M. Gowda. 2014. Affective States and State Tests: Investigating How Affect and Engagement during the School Year Predict End-of-Year Learning Outcomes. *Journal of learning Analytics* 1 (2014), 107–128.

[21] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python . *Journal of Machine Learning Research* 12 (2011), 2825–2830.

[22] J. Elizabeth Richey, Nikki G. Lobczowski, Paulo F. Carvalho, and Kenneth Koedinger. 2020. Comprehensive Views of Math Learners: A Case for Modeling and Supporting Non-math Factors in Adaptive Math Software. In *Artificial Intelligence in Education*, Ig Ibert Bittencourt, Mutlu Cukurova, Kasia Muldner, Rose Luckin, and Eva Millán (Eds.). Springer International Publishing, Cham, 460–471.

[23] Steven Ritter, John R. Anderson, K. Koedinger, and Albert T. Corbett. 2007. Cognitive Tutor: Applied research in mathematics education. *Psychonomic Bulletin & Review* 14 (2007), 249–255.

[24] Steven Ritter, Ambarish Joshi, Stephen Fancsali, and Tristan Nixon. 2013. Predicting Standardized Test Scores from Cognitive Tutor Interactions. In *EDM*.

[25] Steven Ritter, April Murphy, Stephen E. Fancsali, Vivek Fitkariwala, Nirmal Patel, and J. Derek Lomas. 2020. UpGrade: An open source tool to support A/B testing in educational software.. In *Proceedings of the First Workshop on Educational A/B Testing at Scale.* EdTech Books.

[26] Steven Ritter, Michael V. Yudelson, Stephen E. Fancsali, and Susan R. Berman. 2016. How Mastery Learning Works at Scale. *Proceedings of the Third (2016) ACM Conference on Learning @ Scale* (2016).

[27] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* 15, 1 (jan 2014), 1929–1958.

[28] Piia Maria Vilenius-Tuohimaa, Kaisa Aunola, and Jari-Erik Nurmi. 2008. The association between mathematical word problems and reading comprehension. *Educational Psychology* 28, 4 (2008), 409–426. https://doi.org/10.1080/01443410701708228 arXiv:https://doi.org/10.1080/01443410701708228

[29] Michael Waskom, Olga Botvinnik, Drew O'Kane, Paul Hobson, Saulius Lukauskas, David C Gemperline, Tom Augspurger, Yaroslav Halchenko, John B. Cole, Jordi Warmenhoven, Julian de Ruiter, Cameron Pye, Stephan Hoyer, Jake Vanderplas, Santi Villalba, Gero Kunter, Eric Quintero, Pete Bachant, Marcel Martin, Kyle Meyer, Alistair Miles, Yoav Ram, Tal Yarkoni, Mike Lee Williams, Constantine Evans, Clark Fitzgerald, Brian, Chris Fonnesbeck, Antony Lee, and Adel Qalieh. 2017. mwaskom/seaborn: v0.8.1. https://doi.org/10.5281/zenodo.883859

[30] Chuankai Zhang, Yanzun Huang, Jingyu Wang, Dongyang Lu, Weiqi Fang, John C. Stamper, Stephen Fancsali, Kenneth Holstein, and Vincent Aleven. 2019. Early Detection of Wheel Spinning: Comparison across Tutors, Models, Features, and Operationalizations. In *EDM*.

[31] Guoguo Zheng, Stephen Edward Fancsali, Steven Ritter, and Susan Berman. 2019. Using Instruction-Embedded Formative Assessment to Predict State Summative Test Scores and Achievement Levels in Mathematics. *Journal of Learning Analytics* 6, 2 (Aug. 2019), 153–174. https://doi.org/10.18608/jla.2019.62.11

## ACKNOWLEDGMENTS

Figures in this paper were generated using the open-source Python libraries: matplotlib [13] and Seaborn [29]; machine learning models were built and trained using Sci-kit Learn [21], Tensorflow [1], keras [8], and XGBoost [7]; and statistical computations were performed in SciPy [14].

The research reported here was supported, in whole or in part, by the Institute of Education Sciences, U.S. Department of Education, through grant R324A210289 to Center for Applied Special Technology. The opinions expressed are those of the authors and do not represent the views of the Institute or the U.S. Department of Education.

The authors thank Jess Gropen, Kim Ducharme, Rae Bastoni, April Murphy, Susan Berman, Sarah Galasso, Kyle Weldon, Mark Swartz, Mia Gross, Leila Lester, Jennifer Reddig, and Zack Lee for helpful discussions.