Optimizing Human Performance using Individualized Computational Models of Learning

Christopher James MacLellan CHRISTOPHER.MACLELLAN@DREXEL.EDU
College of Computing and Informatics, Drexel University, Philadelphia, PA 19104 USA

Kimberly Stowers KSTOWERS@CBA.UA.EDU
Lisa Brady LLBRADY@CRIMSON.UA.EDU
Culverhouse College of Business, The University of Alabama, Tuscaloosa, AL 35487 USA

Abstract

A key challenge for human performance optimization designers is cost effectively evaluating different interventions. Typically, A/B experiments are used to evaluate interventions, but running these experiments is costly. We explore how computational models of learning can support designers in causally reasoning about alternative interventions for a fractions tutor. We present an approach for automatically tuning models to specific individuals and show that these individualized models make better predictions than generic models. Next, we apply these individualized models to generate counterfactual predictions for how two students (a high and a low-performing student) will respond to three different fractions training interventions. Our model makes predictions that align with previous human findings as well as testable predictions that might be evaluated with future human experiments.

1. Introduction

The goal of human performance optimization interventions is to improve the performance of an individual or team along some desirable dimension. Interventions might target performance on a wide range of tasks, including those that present across the physical, cognitive, or social domains. For example, soldiers might undergo specialized fitness training to improve their physical performance, K12 students might practice solving problems to improve their cognitive performance, and astronauts might get grouped into complementary teams to improve their group performance. Further, interventions can target performance on a wide range of time scales, from seconds and minutes to weeks and months.

Regardless of the task, domain, or timescale, identifying the interventions that best achieve the desired performance goals and evaluating their effectiveness in a cost effective way is a central challenge to human performance optimization decision making. Koedinger et al. (2013) sketch out the design space for educational interventions and claim that those picking an intervention must choose from over 200 trillion possible unique options, even when considering just a small design space with 15 possible instructional techniques, 3 dosage levels, and the possibility of different dosage choices for early and late instruction. How then does one go about evaluating alternative interventions and selecting from this enormous set of possible options in an informed way?

The gold standard in human performance optimization utilizes randomized A/B experiments to evaluate the causal impact of different interventions and to quantify their effectiveness over a baseline ("control") group. Unfortunately, running controlled experiments is a costly endeavour; getting approval for human experimentation and organizing and running experiments is no small task. There are also many limitations of A/B experiments. Experiments often only compare a small number of interventions (typically 2-3) and it is very difficult to generalize from these interventions to other alternatives. The end result is that A/B experimentation reduces human performance optimization to a game of twenty questions with nature (Newell, 1973), where each question is expensive to answer and only provides a single bit of information regarding which intervention is best. Further, A/B experiments often treat interventions as one-size-fits-all solutions, when in reality different interventions often have different effects for different people; e.g., it is a well-known finding that novices learn more from studying worked examples than from engaging in problem solving, but this relationship reverses as students gain more expertise (Kalyuga et al., 2003). Accounting for individual differences typically requires more experimental conditions and an increased cost, but not properly accounting for individual differences when applying interventions hinders potential performance gains.

Given the costs and limitations of A/B experiments, we need computational tools to support teachers, personal trainers, managers, researchers, and other intervention designers in cost effectively picking the best options from the range of possible alternatives. To address this need for those considering cognitive training interventions, we propose the use of computational models of human learning. Similar to how bridge designers use parametric analysis to computationally simulate and test bridges prior to deploying them in the real world, we propose using computational models to simulate and test cognitive training interventions prior to running more costly human experiments. Whereas purely statistical models of human learning (e.g., MacLellan et al., 2015) are very limited in their ability to generalize to interventions without existing human performance data, computational models of learning mechanistically model how a student's knowledge changes in response to an intervention and how their performance changes as a result. By leveraging cognitive learning theories within a unified computational model of learning (Newell, 1994), our previous work suggests that it is possible to make purely theory-driven predictions about human performance for alternative interventions, even when no existing human data are available (Maclellan et al., 2016; MacLellan, 2017).

In addition to providing a means of evaluating counterfactual interventions, computational models of learning also provide a means of addressing the one-size-fits-all problem faced by A/B experiments. Models can be customized to better approximate specific individuals and their unique characteristics (e.g., novices or experts) and predict how these individuals will be uniquely affected by different interventions. Previous work has explored how models can be customized in this way (Jones & VanLehn, 1992; Zhang & Hornof, 2014). To build on these ideas, we explore a novel approach for leveraging performance data when it is available (e.g., from a previous experiment evaluating one possible intervention) to automatically individualize a cognitive model, so it better predicts the performance for the target individual given different possible interventions.

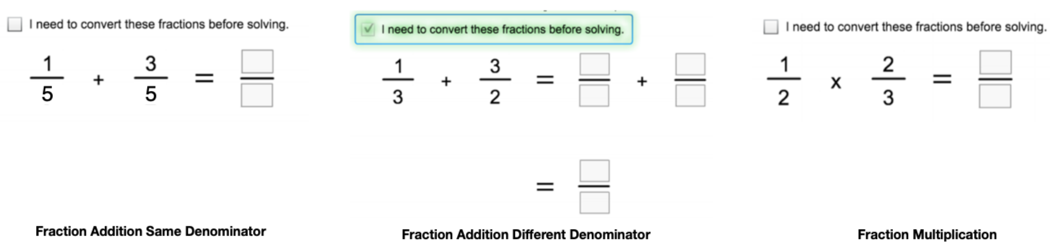


Figure 1. Three types of problems presented in the fractions tutoring system.

In this paper, we explore the use of computational models of learning for supporting human performance optimization intervention design and provide evidence to support three high-level claims:

- Computational models of learning can support causal predictions of human performance and learning for possible interventions, even when prior human data for those interventions is not available;
- These models can be individualized by adjusting their parameters and prior knowledge to better model and predict the performance of specific individuals; and
- Once individualized, they can generate plausible counterfactual predictions for how specific individuals will uniquely respond to different human performance optimization interventions.

To support these claims, we first describe the fraction arithmetic learning environment we used in our modeling effort. We then describe our computational model of human performance and learning on this task. Next, we describe our approach for automatically individualizing cognitive models and present evidence that individualized models better predict human performance than generic, non-individualized models. Finally, we construct individualized models of two students (a high and low-performing student) and use these models to generate plausible counterfactual learning curve predictions for each student across three different human performance optimization interventions. Given that these predictions are counterfactual (Pearl, 2000)—that is, we are making predictions for interventions that were not evaluated in the human data—no ground truth data exists to evaluate them. However, we qualitatively assess their plausibility and show these models generate reasonable predictions that agree with previous findings from human studies.

2. Fraction Arithmetic Tutor

To investigate the use of computational models to support human performance optimization, we chose to model human decision making and learning within a fraction arithmetic tutoring system. Patel et al. (2016) created this tutor to teach students how to solve three types of fraction arithmetic problems: fraction addition with same denominators, fraction addition with different denominators, and fraction multiplication. Figure 1 shows the tutoring system interface for each of these problem types. Following the standard intelligent tutoring system design (Vanlehn, 2006), this tutor provides immediate correctness feedback on each step and students can only proceed once all of the steps have been performed correctly. Additionally, if a student gets stuck, then they can request a "hint" and the tutor provides them with a worked example of how to perform the next step.

The tutor scaffolds students in solving these problems in a particular fashion. For all three problem types, students must decide whether they need to convert the fractions before solving. If they elect to convert and the tutoring system determines that it is appropriate to do so, then they are presented with additional input fields to support conversion (see the middle image of Figure 1). When solving addition with same denominator and multiplication problems, students can input the numerators and denominators in any order and can only mark the problem as done once both fields have correct inputs. When a student is solving an addition problem with different denominators, they must convert the fraction to common denominators before proceeding. In this case, the tutor requires students to use the butterfly method to find common denominators–the two denominators are multiplied to get a common denominator and the opposing numerators and denominators are multiplied to get new numerators. Additionally, students are required to input the converted fraction values in a particular order. First, they must input the lower left denominator, and then they can input either the right denominator or the left numerator, in either order. Finally, the student is can enter the right numerator. Once the fraction has been converted, the answer numerator and denominator can be input in any order. The student can proceed once both the answer fields have correct inputs.[1]

For our analysis we used the "Study 2" data from the publicly available "Fraction Addition and Multiplication" dataset accessed via DataShop (Koedinger et al., 2010). This dataset comes from an experiment conducted by Patel et al. (2016) to investigate whether it is better to block or interleave students' fractions practice. For this experiment, 118 sixth graders were randomly assigned to receive 48 practice problems in either a blocked or interleaved order. Half of the students in the blocked condition received all the addition with same denominators problems, then all the addition with different denominators problems, then all the multiplication problems. The other half received all the multiplication problems, then all the addition with same denominators problems, then all the addition with different denominators problems. For each block, problems were presented in a random order. The students in the interleaved condition received a randomized ordering of all the problems. The main finding of this study was that students have lower error during practice in the blocked condition, but better posttest performance in the interleaved condition, suggesting that interleaving fraction arithmetic practice yields better learning than blocking.

3. Apprentice Learner Architecture

To model human learning in the fractions tutor, we constructed a computational model using the the Apprentice Learner Architecture (Maclellan et al., 2016; MacLellan, 2017), which provides a framework for modeling human learning and decision making from tutoring system interactions. For this work, we created a novel Apprentice Learner model. This model, which is conceptually depicted in Figure 2, has two memories: a long-term memory that contains skills (both hand authored or learned) and a short-term memory that maintains a set of working memory elements that can be manipulated through skill execution. The model uses the Rete algorithm (Forgy, 1989) to efficiently organize skills and to match them against active working memory elements. Skills have a structure

1. We are not theoretically committed to requiring students to use the butterfly strategy or to entering steps in this fixed order, but Patel et al. (2016)'s original human tutor had these requirements and we mirror them in our simulations.

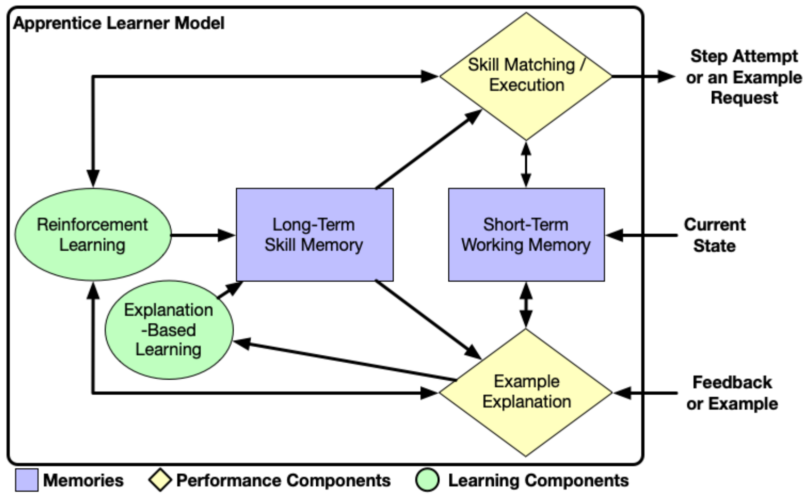


Figure 2. A conceptual depiction of the memories (blue boxes), performance components (yellow diamonds), and learning components (green circles) for our Apprentice Learner model.

common to most production systems: they contain conditions that constrain when the skill applies and effects that update the working memory when executed.

The model has two performance components that it uses to interact with and learn from tutoring system interactions. First, the skill matching / execution component matches skills from long-term memory against the current working memory elements. When multiple skills match, it utilizes a reinforcement learned approximator to estimate which skills will yield the highest future reward and executes those skills first. When the system executes skills that generate external actions, the system attempts a step within the tutoring system. If the system has no actions that produce positive expected reward, then the system requests a hint from the tutor.

When the system receives feedback on its actions or a worked example, it employs its example explanation component. This component applies skills from long-term memory to try and generate the worked example or the last action that triggered feedback. When receiving feedback, the system already has a trace in its working memory to explain the last action (from generating the step initially). When explaining examples, the component uses the same approach as skill matching/execution, but fires skills even if they do not generate a positive expected reward. Once the system has constructed an explanation, it applies explanation-based learning (DeJong & Mooney, 1986) to compile its explanation trace into a new skill, which can then be used in subsequent problem solving and learning. While both performance components are active, the reinforcement learning system uses Deep Q-Learning (Mnih et al., 2015) to continually update the reward approximator after each skill execution. For more details on the Apprentice Learner Architecture and its rationale see (MacLellan, 2017).

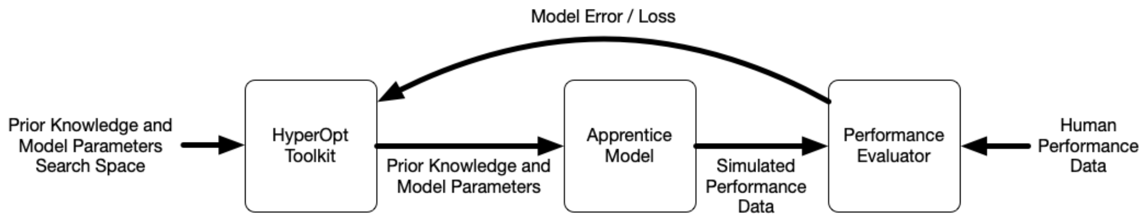


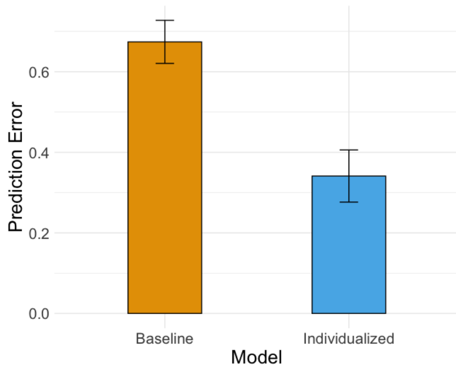
Figure 3. The iterative process used to individualize Apprentice agents.

4. Model Individualization

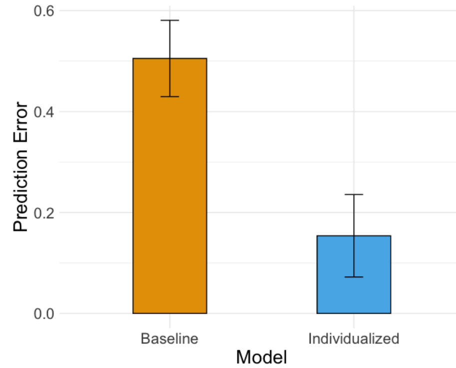
In our prior work (Maclellan et al., 2016; MacLellan, 2017), we explored the use of Apprentice Learner models for predicting which fractions interventions yield the best learning. Our previous results demonstrated that Apprentice Learner models successfully predict that students will have lower error during tutoring in the blocked condition and lower error on a posttest in the interleaved condition. However, when we compared the learning curves generated by the humans and the models, we found a large discrepancy. Our models assume that all students are identical learners and that they come to the fraction arithmetic task without any prior fraction arithmetic knowledge, so they have 100% error on every first opportunity to apply a skill (the model always requests a hint on each first opportunity, which gets counted as an error). In contrast, the data shows that human students make a mistake on their first skill opportunities less than half the time, suggesting that students come into the tutoring system already knowing how to do most of the fraction steps. Beyond prior knowledge differences, we also anticipate that students might have cognitive differences that impact their learning. For example, different students might have varying thresholds for their willingness to guess an answer, which may affect how quickly they learn. Our previous models have not accounted for these individual-level differences.

4.1 Our Individualization Approach

To address this gap, we have created an approach for automatically tailoring Apprentice Learner models to specific students to better account for their differences. Our approach frames model individualization as a hyperparameter optimization problem and uses the HyperOpt tookit (Bergstra et al., 2013) to automate the process of individualizing our models, see Figure 3. Each Apprentice Learner model accepts a set of hyperparameters that define the skills a model starts with as well as some additional cognitive parameters that are used by the reinforcement learning system. To utilize HyperOpt, we provide it with a specification of the space of model prior knowledge and cognitive parameters. To individualize a model to a specific student, HyperOpt iteratively samples a set of hyperparameters from the provided hyperparameter space. For each set of hyperparameters, it creates an Apprentice Learner model that utilizes these parameters, simulates the target student and the intervention that they received using the model, and evaluates how accurately the model emulates to the target student's behavior on each step. The error between the model and the human is fed back to the HyperOpt toolkit, which uses Bayesian inference to update its hyperparameter sampling



(a)



(b)

Figure 4. (a) Prediction Error for Apprentice Learner Models that are trained and tested on the same data. (b) Prediction Error for Apprentice Learner Models that are trained on earlier data and tested on later unseen data (within individuals). In both cases, the error bars represent the 95% confidence intervals.

distribution. With each iteration, HyperOpt converges to the hyperparameters that minimize the error between the model and the target student.

To apply this approach to individualizing our models for the fraction arithmetic tutor, we created a set of prior knowledge that our models could potentially start with. In our previous work all models had prior knowledge for whole number arithmetic (adding, subtracting, multiplying, and dividing two numbers). For this work, our prior knowledge space includes these skills as well as all the correct fraction arithmetic skills for: adding fractions with same denominators, converting fractions to common denominators with the butterfly method, and multiplying fractions. If an agent starts with only the correct fraction arithmetic skills it will get every step correct, even without any practice. However, if it has only a subset of these skills, then it will get some steps correct and have to learn other steps through worked examples and practice. Our models also have hyperparameters for the reinforcement learning system that include how often it guesses random actions (epsilon), how much of a reward penalty the agent gets for taking an action (action_penalty, to minimize unnecessary actions), and how much reward is decayed when propagating it back over each step (gamma).

4.2 Individualization Evaluation

We conducted two evaluations of our approach. First, we evaluated whether HyperOpt is able to successfully identify hyperparameters that improve the alignment between an Apprentice Learner model and a target human. We individualized 24 Apprentice Learner models to 24 students using 20 iterations of HyperOpt optimization in each case. For the performance evaluation step of our individualization process, see Figure 3, we simulated each student's behavior on the first ten problems that they received in the tutor and computed the model error as the difference between first attempt correctness of the models and humans on each step. After fitting a model to each student, we compared the individualized models to a baseline model that had only whole number arithmetic prior knowledge (no prior fraction arithmetic knowledge) and default cognitive parameters (epsilon= 0.3, gamma= 0.7, action_penalty= 0.05). This baseline model corresponds to the configuration used in our previous work (Maclellan et al., 2016; MacLellan, 2017). To evaluate each model, we computed the first attempt correctness prediction error between each model and their respective student on the first 10 problems they received in the tutor. Figure 4(a) shows the results of this evaluation. Our main finding is that the individualized models have lower error than the baseline models and better approximate the human learning trajectories.

This first evaluation demonstrates that our individualization approach reduces the error between the models and the specific students. However, it evaluates the model on the same tutor problems that were used to perform the model individualization–effectively training and testing on the same data. To evaluate how well the individualized models are able to make improved predictions for the target students on unseen data, we utilized a form of temporal cross validation. We constructed individualized models for 15 students by using HyperOpt to reduce the error between the models and humans on the first 5 problems they practiced in the tutor. We then compared these individualized models to our baseline model on the next 15 problems that they received within the tutor (problems 6 through 20), which were not used as part of the model individualization process. Figure 4(b) shows the results of this evaluation. The main conclusion we can draw from this second evaluation is that the individualized models better approximate the human behavior even on unseen data that was not used as part of the model individualization process. It is worth noting that the error of the models in this second evaluation are lower than the first because students' error tends towards 0% as they receive more fractions practice, so predicting the human performance becomes easier at later opportunities.

5. Individualized Counterfactual Prediction

Although Patel et al. (2016) argue for a one-size-fits all approach to problem ordering–that interleaved practice yields better learning than blocked practice–we take a more nuanced view that different kinds of practice might be better for different individuals, depending on their prior knowledge as well as other individual cognitive differences. Unfortunately, this view complicates the human performance optimization design problem because it means that not only does a designer need to find the single best intervention, but is instead faced with finding the best intervention for each individual. Having demonstrated that our model individualization approach can successfully tailor models to make improved predictions for specific students, we next explored how these individualized models might be leveraged to inform the selection of which tutoring interventions might be best for each student, providing a solution to this more complicated intervention design problem.

To support designers, our models can generate counterfactual predictions of students' performance. If a designer has access to previously collected intervention data for a student, then they can individualize Apprentice Learner models to that student using this data. They can then use this individualized model to counterfactually predict what the student's performance would have been had they received a different condition than the one they actually received. It is also possible for designers to individualize a model using other kinds of performance data, such as pretest performance data. By leveraging these other kinds of data, a designer can generate counterfactual predictions

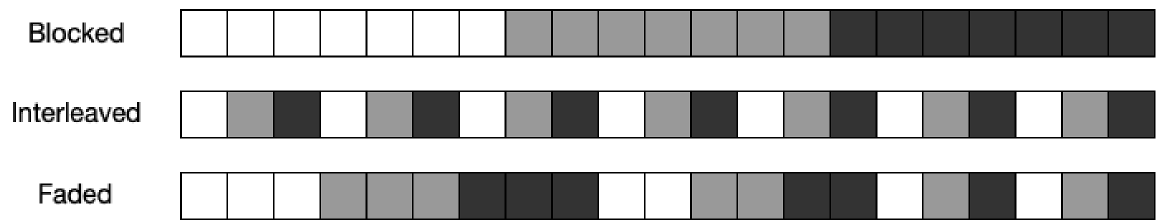


Figure 5. Conceptual depiction of the three fraction arithmetic problem ordering interventions. Different shadings of cells represent the three different types of problems: fraction addition with same denominators, fraction arithmetic with different denominators, and fraction multiplication.

about how a particular student would respond to an intervention prior to actually administering any interventions to them.

For this work, we explore the use of our models for supporting counterfactual predictions. In particular, we explore how our Apprentice Learner models can be applied to answer four counterfactual questions regarding the students that used the fraction arithmetic tutor:

- Q1: What would the learning curves look like if a participant in the interleaved condition had received the blocked condition instead, and vice versa?
- Q2: What would have happened if a participant received a novel intervention where the problems were presented in a fashion that started as blocked but faded to interleaved over the course of the instruction?
- Q3: What would a low-performing student's learning curves look like for each intervention?
- Q4: What would a high-performing student's learning curves look like for each intervention?

The first of these questions explores what would have happened if a student received a different condition from the one they were randomly assigned to. The second question investigates how the interventions evaluated in the human study might compare to a novel faded intervention for which no data are currently available. This novel intervention starts out like the blocked instruction, but then fades into interleaved instruction over the course of instruction. Figure 5 provides a conceptual depiction of the three different instructional interventions. We chose to explore this faded blocked to interleaved intervention because previous work (Carvalho & Goldstone, 2015) shows that blocking and interleaving support different kinds of learning—blocking helps students learn which task features are relevant but interleaving helps student learn to discriminate among competing skills. Finally, the third and fourth questions explore how the alternative interventions might deferentially affect high vs. low-performing students. In general, we feel these counterfactual questions are representative of the kinds of questions a designer might have.

5.1 Our Counterfactual Prediction Approach

Our counterfactual approach consists of four steps. First, we individualize our models by tuning them to each target individual using their performance data from the intervention they received. Next, we simulate the counterfactual interventions to determine what the respective student might have done had they received an alternative intervention. Finally, we analyze the simulated performance data to determine what the model predicts would happen in the counterfactual interventions.

5.2 Counterfactual Prediction Evaluation

To evaluate our approach, we applied it to answering our four counterfactual questions. We focused our simulation efforts on two students from human data: a high-performing and a low-performing student. The high-performing student had the highest tutor performance of any student in the blocked condition and the low-performing student had the lowest tutor performance of any student in the interleaved condition. For each student, we constructed an individualized Apprentice Learner model using their available performance data. We chose to evaluate our model on two specific students rather than all of the students because we wanted to understand how well the model predicts specific individuals rather than aggregate performance as well as how model predictions differ after individualizing to different kinds of students.

After constructing individualized models, we applied them to simulate three different counterfactual interventions for each student. First, we simulated the students in different variations of the condition they actually received (blocked or interleaved, where each variation has a different randomized problem ordering within the target ordering schema). By comparing the simulated behavior on these different variations to the actual student performance, we can get a sense of how well our model predicts the student's learning trajectory for the observed condition. Next, we simulated student performance on the opposite condition from the one they received. Finally, we simulated their behavior on the novel faded blocked to interleaved condition. For each counterfactual simulation, we simulated the student behavior within that condition 20 times. For each iteration in the blocked and interleaved conditions, we randomly selected a problem ordering from the set of sequences that were actually administered to people in that condition from the larger dataset. In the case of the faded condition, we randomly generated faded sequences where students received problems in blocks of three (three addition with same denominators, then three addition with different denominators, then three multiplication), then blocks of two, and eventually blocks of one. Across these three evaluations, our models generated predictions for interventions that differed from those actually experienced by the humans, so no ground truth was available (i.e., they are counterfactual predictions).

5.2.1 High-Performing Student Predictions

Figure 6 shows the observed and simulated learning curves for the high-performing student as they interact with the fraction arithmetic tutor in each of the three interventions (blocked, interleaved, and faded). The learning curves show the average performance across all fraction arithmetic skills. The opportunity count represents how many prior opportunities the student has had to exercise each fraction arithmetic skill. Thus, the error at opportunity zero corresponds to the average performance the first time all fraction arithmetic skills are applied within the tutor. For these plots, we used a skill model that maps each input field for each problem type to a unique skill (e.g., inputting the answer numerator for an addition with same denominators problem uses a different skill than inputting the answer numerator for a multiplication problem). Additionally, each plot includes an estimate from

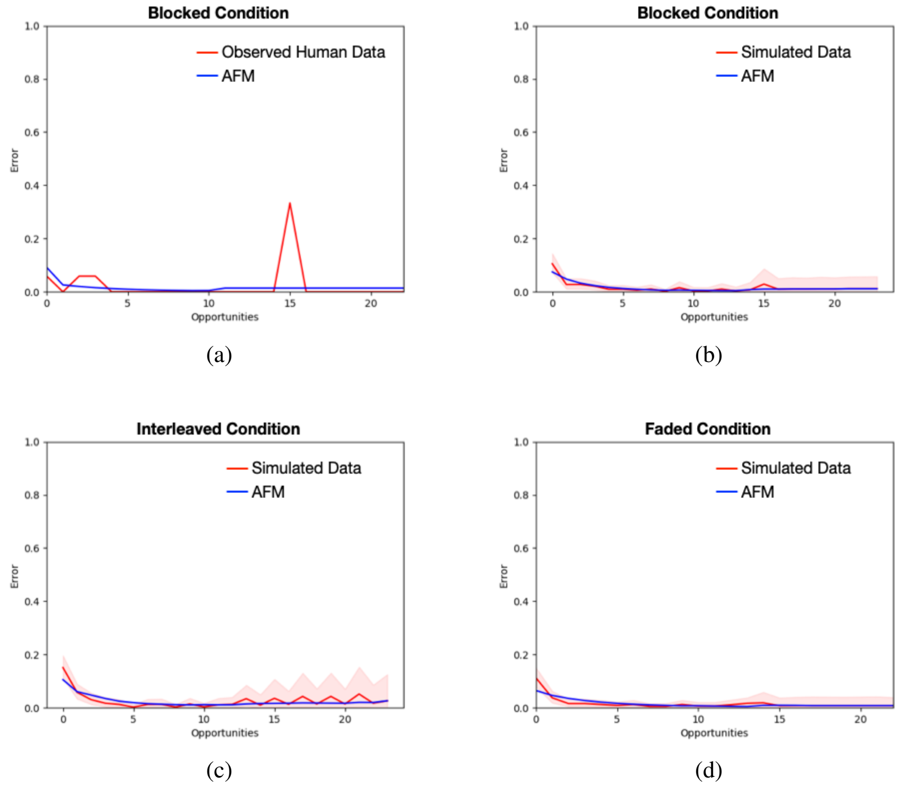


Figure 6. (a) The high-performing student's actual performance in the blocked condition. The predicted performance for the high-performing student across variations of the (b) blocked condition, (c) interleaved condition, and (d) faded blocked to interleaved condition. The shaded regions denote the 95% confidence intervals for predicted error at each opportunity and the blue lines represent the estimates from an additive factors model (AFM) fit to the data in each case.

the best fitting additive factors model (Cen et al., 2006), which represents a constrained logistic regression curve that accounts for how performance improves with practice.

If we compare the actual human performance, Figure 6(a), and our model prediction for how the student would perform in variations of that condition, Figure 6(b), we find that the model does a reasonable job of emulating the observed human performance. In particular, there is not a large discrepancy between error rates on the first opportunity, which has been observed in prior Apprentice Learner work. Previous models had 100% error rate on the first opportunity because they always started without any fraction arithmetic knowledge and were not individualized to specific students (Maclellan et al., 2016; MacLellan, 2017; Weitekamp III et al., 2019). Our results suggest we can overcome for the disagreement between our prior models and humans by taking into account individual student's prior knowledge.

Additionally, we see that our model correctly predicts a spike in error rate around opportunity 15, likely due to a transition between problem blocks around this opportunity.[2] If we look at the predicted performance in the interleaved condition, Figure 6(c), we see an interesting spiky pattern in error in the tail of the learning curve as students alternate between problems of different types. Our model also predicts that on average the error in the interleaved condition will be higher than in the blocked condition. This prediction agrees with Patel et al. (2016)'s finding the human students make more errors within the tutor in the interleaved condition than the blocked condition. When looking at the predicted performance in the faded condition, Figure 6(d), we find that performance looks almost identical to the blocked condition, even though the problems are essentially interleaved at the higher opportunity counts. Interestingly, our model does not predict spikes in error like it did for the interleaved condition. This suggests that perhaps the faded condition successfully combines the benefits of both blocking and interleaving, achieving better learning and perhaps better transfer to a posttest, which we did not evaluate.

5.2.2 Low-Performing Student Predictions

Similar to the high-performing student results, Figure 7 shows the observed and simulated learning curves for the low-performing student as they interact with the fraction arithmetic tutor in each of the three possible interventions (blocked, interleaved, and faded). Our first general observation is that the low-performing student makes many more errors than the high-performing student. Still, the student gets approximately half of their first steps correct, suggesting that they already have a fair amount of fractions knowledge prior to using the tutor.

If we compare the observed performance for the interleaved condition, 7(a), to the predicted performance for variations of the interleaved condition, Figure 7(b), we find that the model does a reasonable job of predicting the human performance. There is only a minor difference in first opportunity error, with the model predicting a slightly higher first opportunity error. As mentioned previously, this discrepancy is not as large as has been observed in previous Apprentice Learner work, where the model always predicts 100% error on the first step. Additionally, the observed human performance is generally within the predicted confidence intervals generated by the model. If we compare, the predicted interleaved performance, Figure 7(b), to the predicted blocked performance, Figure 7(c), we see that the model predicts that the tutored problem error rates in the blocked condition will be lower than the interleaved condition, which agrees with the predictions for the high-performing student as well as the general finding from the human data (Patel et al., 2016). Lastly, the predicted performance for the faded condition, Figure 7(d), shows lower overall error and a faster decrease in error than either the interleaved or blocked conditions. In particular, the error in the faded condition decreased to approximately 20% by opportunity 3, where this level of error is not achieved in the interleaved condition until opportunity 15 or the blocked condition until opportunity 5. We did not find the same improvement for the high-performing student in the faded condition, but they had less range to improve. This finding further suggests that the faded condition successfully combines the benefits of both blocking and interleaving.

2. Figure 6(a) shows data from just a single human student on a single sequence whereas the simulated data were computed over 20 students on 20 sequences; this is why it has no error bars and the error rate on opportunity 15 appears to be much higher.

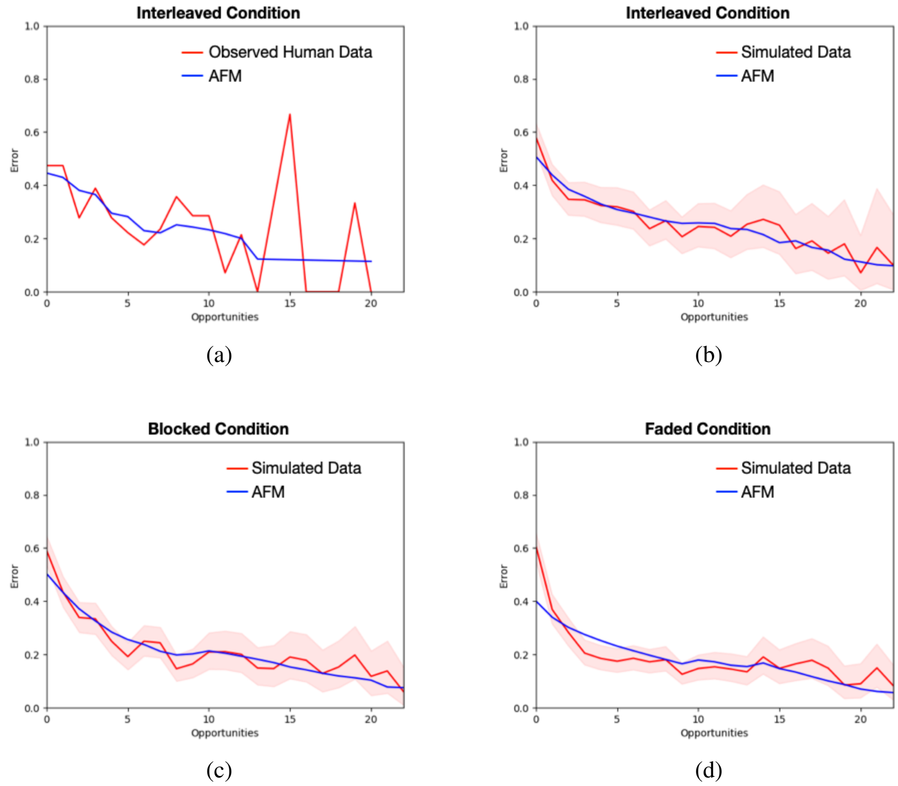


Figure 7. (a) The low-performing student's actual performance in the interleaved condition. The predicted performance for the low-performing student across variations of the (b) interleaved condition, (c) blocked condition, and (d) faded blocked to interleaved condition. The shaded regions denote the 95% confidence intervals for predicted error at each opportunity and the blue lines represent the estimates from an additive factors model (AFM) fit to the data in each case.

5.3 Discussion

These results demonstrate how our Apprentice Learner models can be used by intervention designers to make individualized counterfactual predictions about how a specific student will respond to possible interventions. This capability is powerful because it will enable designers to conduct low cost simulations to evaluate many alternative interventions. They can then run human studies to evaluate the interventions that the simulations suggest will be the most promising. However, before designers can trust Apprentice Learner model predictions, they need some evidence that the counterfactual predictions reasonably approximate human performance. Unfortunately, evaluating counterfactual predictions is difficult because, by definition, no ground truth data are available to compare against.

We claim that our findings provide preliminary evidence to suggest that our models can make reasonable counterfactual predictions about how different individuals will respond to alternative interventions. Across both the high and low-performing students, we found that our model predictions seemed reasonable. When we compare the observed human performance to the model's predictions for variations of the observed condition, we see a close agreement between model and humans. Further, across both students our model predicts that tutored problem error will be lower in the blocked condition than in the interleaved condition, which has been observed to be true in the human data (Patel et al., 2016).

It is harder to evaluate the model's predictions regarding the faded condition because no human performance data are available to validate our model's predictions. However, our model makes a reasonable, but not entirely obvious, prediction that students in the faded condition will have lower overall error and that their error will decrease more quickly than either the blocked or interleaved conditions. Additionally, the model predicts that in the tail of the learning curve the error will look more like the blocked condition (less spiky) than the interleaved condition (more spiky), even though practice at higher opportunities in the faded condition is essentially interleaved. Despite being unable to currently validate these predictions with human data, we argue that they constitute reasonable counterfactual predictions that do not disagree with prior research on blocking vs interleaving (Carvalho & Goldstone, 2015). A good future test of the Apprentice Learner models would be to run a human study comparing this faded problem ordering to the blocked and interleaved conditions to see if our model predictions are substantiated.

Finally, these simulated counterfactual data provide some answers to our four counterfactual questions. They give us a picture of how students' learning curves might differ if they were in the blocked vs. interleaved condition (Q1). They show us how the students' learning curve might differ if they were in a novel faded condition (Q2). Finally, they provide us with a picture of how low-performing and high-performing students would respond differently to these three interventions (Q3 and Q4). In general, our models seem to suggest that which intervention the high-performing student receives almost does not matter because they already have a very good knowledge of how to solve fraction arithmetic problems prior to using the tutor. However, the low-performing student improves in all three conditions, but seems to improve the most in the faded condition.

6. Related Work

Our work is not the first to explore the application of computational models to guiding the design of interactive systems. For example, the early work of Card et al. (1986) proposed the use of a Model Human Processor that encapsulates cognitive theory into a computational model for evaluating the usability of interface designs in lieu of more costly human A/B experiments. More recently John et al. (2004) have worked to realize this vision through the development of the CogTool system that supports designers in building usable interfaces. We view our work as building on these past efforts and extending them in at least two respects. Our work explores how computational models can support the design of human performance optimization interventions rather than interface designs. Additionally, our work centers around modeling learning and how human performance changes as they use the system, whereas these related efforts focuses primarily on modeling decision making and how usable an interface will be after a person has been already been trained in how to use it.

There are also many related efforts that investigate how to individualize cognitive models to specific people. For example, early work by Jones & VanLehn (1992) explored how researchers can hand tune the prior knowledge and parameters of cognitive models to align them with protocols from specific students. A more recent effort by Zhang & Hornof (2014) uses large-scale simulation of all the possible prior knowledge and parameter models to identify those configurations that most closely approximate (and explain) specific individuals behaviors and strategies. Weitekamp III et al. (2019) tune Apprentice Learner models to specific students using a more implicit approach. Rather than searching over the space of prior knowledge directly, they statistically estimate how much previous practice each student has had with each type of problem and pretrain Apprentice Learner models on an equivalent number of comparable problems. Our work builds on these prior efforts to tune our models' prior knowledge and parameters to the target student learning protocols. We chose to take an explicit approach and estimate prior knowledge directly rather than the amount of previous practice. We also chose to use an automated tuning approach rather than tuning the models by hand. However, rather than simulating all possible configurations we leveraged recent developments in hyperparameter optimization (Bergstra et al., 2013) to more efficiently search the space of prior knowledge and parameters. More work is needed to compare how an explicit approaches to tuning models compare to the implicit approach suggested by Weitekamp III et al. (2019). A nice feature of our explicit approach is that it generates a human interpretable set of prior knowledge (the designer can see what prior knowledge our system estimates that a student starts with). However, our approach requires designers to hand construct the set of prior knowledge to search over, whereas the implicit approach does not have this requirement.

7. Conclusions and Future Work

This paper presents evidence to support three high-level claims: (1) that computational models of learning can support causal prediction of human performance and learning in possible interventions even when human data for those interventions is not available, (2) that these models can be individualized by adjusting their model parameters and prior knowledge to better predict performance for specific individuals, and (3) that individualized cognitive models can generate plausible counterfactual predictions for their target individuals. To support these claims, we built a model using the Apprentice Learner Architecture and demonstrated its use for causally reasoning about which fraction arithmetic tutoring interventions will produce better learning in specific students. We described how to individualize these models using available performance data and showed that individualized models better predict performance than generic models. Finally, we constructed individualized models for one high-performing and one low-performing student and used these models to counterfactually predict their learning curves for three different fraction arithmetic interventions. Our results show that the individualized models make reasonable predictions that agree with the available human data. They also generate plausible predictions regarding a novel intervention for which no human data are available that might be tested with future human experiments.

To build on this work, there are many possible directions for future work. We are particularly interested in preregistering our model's predictions regarding our proposed faded blocked to interleaved intervention and running a human experiment to see if the predictions are substantiated. Additionally, our current and past modeling efforts have centered primarily around predicting performance and learning in tutors, but future work should explore how our approach can be applied to other learning environments, such as educational games. We are interested in expanding our models to account for additional learning phenomena, such as the testing effect, where students' performance improves after taking tests even though they do not get any feedback or instruction during test taking. With the ability to evaluate the effectiveness of different interventions using Apprentice Learner models, we would also like to explore approaches for automatically searching the space of training interventions; e.g., searching over the space of alternative problem orderings to find those that yield the best learning. Finally, future work should explore how to make these models more accessible to human performance optimization intervention designers, such as teachers or instructional designers.

Acknowledgements

The first author started this work at Soar Technology Inc. under the DARPA TAILOR program (award no. HR00111990055) and then finished the work at Drexel University. The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government. We thank Rob Sheline, Ben Mastay, and Frank Koss their technical development support.

References

Bergstra, J., Yamins, D., & Cox, D. D. (2013). Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In S. Dasgupta & D. McAllester (Eds.), *Proceedings of the 30th international conference on machine learning*. JMLR.

Card, S., MORAN, T., & Newell, A. (1986). The model human processor- an engineering model of human performance. *Handbook of perception and human performance.*, *2*.

Carvalho, P. F., & Goldstone, R. L. (2015). The benefits of interleaved and blocked study: Different tasks benefit from different schedules of study. *Psychonomic bulletin & review*, *22*, 281–288.

Cen, H., Koedinger, K., & Junker, B. (2006). Learning factors analysis–a general method for cognitive model evaluation and improvement. In M. Ikeda, K. Ashlay, & T.-W. Chan (Eds.), *Proceedings of the 8th international conference on intelligent tutoring systems*, 164–175. Springer.

DeJong, G., & Mooney, R. (1986). Explanation-based learning: An alternative view. *Machine learning*, *1*, 145–176.

Forgy, C. L. (1989). Rete: A fast algorithm for the many pattern/many object pattern match problem. In *Readings in artificial intelligence and databases*, 547–559. Elsevier.

John, B. E., Prevas, K., Salvucci, D. D., & Koedinger, K. (2004). Predictive human performance modeling made easy. In A. Monk & D. Olsen (Eds.), *Proceedings of the sigchi conference on human factors in computing systems*, 455–462. ACM.

Jones, R. M., & VanLehn, K. (1992). A fine-grained model of skill acquisition: Fitting cascade to individual subjects. In J. K. Kruschke (Ed.), *Proceedings of the fourteenth annual conference of the cognitive science society*, 873–878. Psychology Press.

Kalyuga, S., Ayres, P., Chandler, P., & Sweller, J. (2003). Expertise reversal effect. *Educational Psychologist*, *38*, 23–31.

Koedinger, K. R., Baker, R. S., Cunningham, K., Skogsholm, A., Leber, B., & Stamper, J. (2010). A data repository for the edm community: The pslc datashop. *Handbook of educational data mining*, *43*, 43–56.

Koedinger, K. R., Booth, J. L., & Klahr, D. (2013). Instructional complexity and the science to constrain it. *Science*, *342*, 935–937.

MacLellan, C. J. (2017). *Computational models of human learning: Applications for tutor development, behavior prediction, and theory testing*. Doctoral dissertation, Carnegie Mellon University.

Maclellan, C. J., Harpstead, E., Patel, R., & Koedinger, K. R. (2016). The apprentice learner architecture: Closing the loop between learning theory and educational data. In T. Barnes, M. Chi, & M. Feng (Eds.), *Proceedings of the 9th international conference on educational data mining*, 151–158. International Educational Data Mining Society.

MacLellan, C. J., Liu, R., & Koedinger, K. R. (2015). Accounting for slipping and other false negatives in logistic models of student learning. In O. C. Santos, J. G. Boticario, M. Pechenizkiy, A. Merceron, P. Mitros, J. M. Luna, C. Mihaescu, A. Moreno, A. Hershkovitz, S. Ventura, & M. Desmarais (Eds.), *Proceedings of the 8th international conference on educational data mining*, 151–158. International Educational Data Mining Society.

Mnih, V., et al. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*, 529–533.

Newell, A. (1973). You can't play 20 questions with nature and win: Projective comments on the papers of this symposium. In W. C. Chase (Ed.), *Visual information processing*. Academic Press.

Newell, A. (1994). *Unified theories of cognition*. Harvard University Press.

Patel, R., Liu, R., & Koedinger, K. R. (2016). When to block versus interleave practice? evidence against teaching fraction addition before fraction multiplication. In A. Papafragou, D. Grodner, D. Mirman, & J. C. Trueswell (Eds.), *Proceedings of the 38th annual conference of the cognitive science society.*, 2069–2074. Cognitive Science Society.

Pearl, J. (2000). Causality: Models, reasoning and inference cambridge university press. *Cambridge, MA, USA,*, *9*, 10–11.

Vanlehn, K. (2006). The behavior of tutoring systems. *International journal of artificial intelligence in education*, *16*, 227–265.

Weitekamp III, D., Harpstead, E., MacLellan, C. J., Rachatasumrit, N., & Koedinger, K. R. (2019). Toward near zero-parameter prediction using a computational model of student learning. In C. F. Lynch, A. Merceron, M. Desmarais, & R. Nkambou (Eds.), *Proceedings of the 12th international conference on educational data mining*, 456–461. International Educational Data Mining Society.

Zhang, Y., & Hornof, A. J. (2014). Understanding multitasking through parallelized strategy exploration and individualized cognitive modeling. In R. Grinter, T. Rodden, P. Aoki, E. Cutrell, R. Jeffries, & G. Olson (Eds.), *Chi '06: Proceedings of the sigchi conference on human factors in computing systems*, 3885–3894. ACM.