Integrating Latent-Factor and Knowledge-Tracing Models to Predict Individual Differences in Learning

Mohammad M. Khajah, Rowan M. Wing, Robert V. Lindsey, Michael C. Mozer
Department of Computer Science and Institute of Cognitive Science
University of Colorado, Boulder, CO 80309-0430
{mohammad.khajah,rowan.wing,robert.lindsey,mozer}@colorado.edu

ABSTRACT

An effective tutor—human or digital—must determine what a student does and does not know. Inferring a student's knowledge state is challenging because behavioral observations (e.g., correct vs. incorrect problem solution) provide only weak evidence. Two classes of models have been proposed to address the challenge. Latent-factor models employ a collaborative filtering approach in which data from a population of students solving a population of problems is used to predict the performance of an individual student on a specific problem. Knowledge-tracing models exploit a student's sequence of problem-solving attempts to determine the point at which a skill is mastered. Although these two approaches are complementary, only preliminary, informal steps have been taken to integrate them. We propose a principled synthesis of the two approaches in a hierarchical Bayesian model that predicts student performance by integrating a theory of the temporal dynamics of learning with a theory of individual differences among students and problems. We present results from three data sets from the DataShop repository indicating that the integrated architecture outperforms either alone. We find significant predictive value in considering the difficulty of specific problems (within a skill), a source of information that has rarely been exploited.

Keywords

Bayesian knowledge tracing, cognitive modeling, collaborative filtering, latent factor models, hierarchical Bayesian models

1. INTRODUCTION

Intelligent tutoring systems (ITS) employ cognitive models to track and assess student knowledge. Beliefs about what a student knows and doesn't know allow an ITS to dynamically adapt its feedback and instruction to optimize the depth and efficiency of learning. A student's knowledge *state* can be described by the specific concepts and operations that have been mastered in the domain of study. These atomic elements are often referred to as *knowledge components* or *skills*. (We use the latter term.) For example, in a geometry curriculum, the *parallelogram-area* skill involves being able to compute the area of a parallelogram given the base and height [6]. Solving any problem typically requires breaking the problem into a series of *steps*, each requiring the application of one or more skills. For example, solving for x in $3(x+2) = 15$ might be broken down into two steps: (1) *eliminate-parentheses*, which transforms $3(x+2) = 15$ to $x+2 = 5$, and (2) *remove-constant*, which simplifies $x+2 = 5$ to $x = 3$ [14]. Because the terminology 'problem step' is cumbersome, we shorten it to 'problem' in the rest of this paper.

A key challenge in student modeling is predicting a student's success or failure on each problem. Following a common practice in the literature, we focus on modeling performance on individual skills. Formally, for a particular skill, the data consist of a set of binary random variables indicating the correctness of response on the i'th problem attempted by a student s, $\{X_{si}\}$. The data also include the problem labels, $\{Y_{si}\}$, which provide a unique index to each problem in the ITS. Recent work has considered secondary data, including the student's utilization of hints, response time, and characteristics of the specific problem and the student's particular history with the problem [2, 27]. Although such data improve predictions, the bulk of research in this area has focused on the primary success/failure data, and a sensible research strategy is to determine the best model based on the primary data, and then to determine how to incorporate secondary data.

2. EXISTING MODELS OF STUDENT LEARNING AND PERFORMANCE

The challenge inherent in predicting student performance is that knowledge state is a hidden variable and must be inferred from patterns of student behavior. Due to the intrinsic uncertainty associated with the inference problem, past approaches have been probabilistic in nature. Two broad classes of approaches have been explored, which we'll refer to as *latent-factor models* and *Bayesian knowledge tracing*, and some preliminary efforts have been made to synthesize the two. In this paper, we present a principled Bayesian unification of the two classes of models. We begin, however, with a summary of past work.

2.1 Latent-factor model

Traditional psychometric methods such as item-response theory [11] use data from a population of students solving a common set of problems to infer the latent ability of each student and the latent difficulty of each problem. These methods can be used to predict student performance. The simplest such model supposes that the log odds of a correct response by student s on trial i is given by $\text{logit}[P(X_{sy} = 1|Y_{si} = y)] = \alpha_s - \delta_y$, where, as before, Y_{si} denotes the problem index, α_s denotes the student's ability and δ_y denotes the problem's difficulty. We refer to this model class as *latent-factor models* or *LFMs*. The left panel of Figure 2 summarizes a Bayesian LFM in graphical model form, with priors on the abilities and difficulties (details to follow shortly), and with $G \equiv P(X_{sy} = 1|Y_{si} = y)$.

Latent-factor models have been used within the ITS community to characterize student performance and predict the consequences of instructional interventions. Examples include performance factors analysis [23], learning factors analysis [6, 5], and instructional factors analysis [8]. Although these models incorporate a wide range of factors, only a few papers have considered what has historically been at the core of latent-factor models, the difficulty of a specific problem. Consequently, a *remove-constant* problem step that simplifies $x + 1 = 3$ is typically considered to to be equivalent to problem step that simplifies $x + 8 = 11$.

2.2 Bayesian knowledge tracing

Bayesian knowledge tracing (*BKT*) [9] is based on a theory of all-or-none human learning [1], which postulates that the knowledge state of student s following trial i, K_{si}, is binary: 1 if the skill has been mastered, 0 otherwise. BKT, often conceptualized as a hidden Markov model, infers K_{si} from the sequence of observed responses on trials $1 \ldots i$, $\{X_{s1}, X_{s2}, \ldots, X_{si}\}$. Table 1 presents the model's four free parameters.

Because BKT is typically used in modeling practice over brief intervals, the model assumes no forgetting, i.e., K cannot transition from 1 to 0. This assumption greatly constrains the time-varying knowledge state: it must make at most one transition from $K = 0$ to $K = 1$ over the sequence of trials. Denoting the trial following which the transition is made as τ, the generative model specifies:

$$P(\tau = i) = \begin{cases} L_0 & \text{if } i = 0 \\ (1 - L_0)T(1 - T)^{i-1} & \text{if } i > 0 \end{cases}$$

$$P(X_{si} = 1|G, S, \tau) = \begin{cases} G & \text{if } i \leq \tau \\ 1 - S & \text{otherwise} \end{cases}$$

The middle panel of Figure 2 shows a graphical model depiction of BKT with the knowledge-state transition sequence represented by τ. With this representation, marginalization over τ is linear in the number of trials, permitting the efficient computation of the posterior predictive distribution, $P(X_{s,i+1} \mid X_{s1}, \ldots, X_{si})$.

2.3 Prior efforts to unify latent-factor and knowledge-tracing models

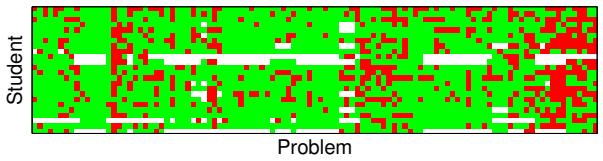

Figure 1: Student × problem matrix for the *Geometry Area* data, obtained from the PSLC DataShop [17]. Correct and incorrect responses are green and red, respectively; white indicates missing data. Students who attempted few problems have been omitted.

Latent-factor and knowledge-tracing models have complementary strengths and weaknesses. LFM addresses individual differences among students and problems. However, because it does not consider the order in which problems are solved, it ignores the likely possibility that performance improves over practice. BKT characterizes the temporal dynamics of learning. However, because it makes no distinction among students or problems, it ignores confounding factors on performance. A natural extension of the models is to formulate some type of combination that yields a more robust representation of knowledge state.

Interesting extensions have been proposed to each model to move it toward the other. Starting with LFM, the latent factors have been augmented with non-latent factors that represent facets of study history such as the amount and success of past practice and the type of instructional intervention [5, 6, 7, 8, 19, 23]. However, these approaches reduce the specific sequential ordering of problems to a few summary statistics, which may not be sufficient to encode the relevant history of past experience.

Many proposals have been put forth to adapt parameters of BKT to individual students. The original BKT paper [9] included heuristic parameter adjustments based on the initial trials in the problem sequence. Another heuristic approach involves the contextualization of guess and slip probabilities based on a range of features such as help requests, response time history, and ITS history [2, 10]. The initial mastery parameter L_0 has been individualized to students, based both on their performance on other skills [20] and on an inferred latent ability parameter [26]. Rather than adapting parameters to individual students, [22] clustered students based on their ITS usage patterns and fit separate parameters for each cluster. The latter two methods require previous history with a particular student, though placing Dirichlet priors on guess and slip rates [3, 4] has been used not only to individuate the parameters for a particular student but to allow for generalization to new students.

Most applications of BKT fit model parameters independently for each skill. There are only a few examples of modulating parameters based on the specific problem being solved. In [13], problem difficulty is represented by using the average number of correct responses on a problem as a feature in the contextualization model of [2]. In the KT-IDEM model [21]—the work closest to our own—the guess and slip parameters are fit individually for each problem within a skill.

Table 1: Free parameters of BKT

L_0	$P(K_{s0} = 1)$	probability that student has mastered skill prior to solving the first problem
T	$P(K_{s,i+1} = 1 \mid K_{si} = 0)$	transition probability from the not-mastered to mastered state
G	$P(X_{si} = 1 \mid K_{si} = 0)$	probability of correctly *guessing* the answer prior to skill mastery
S	$P(X_{si} = 0 \mid K_{si} = 1)$	probability of answering incorrectly due to a *slip* following skill mastery

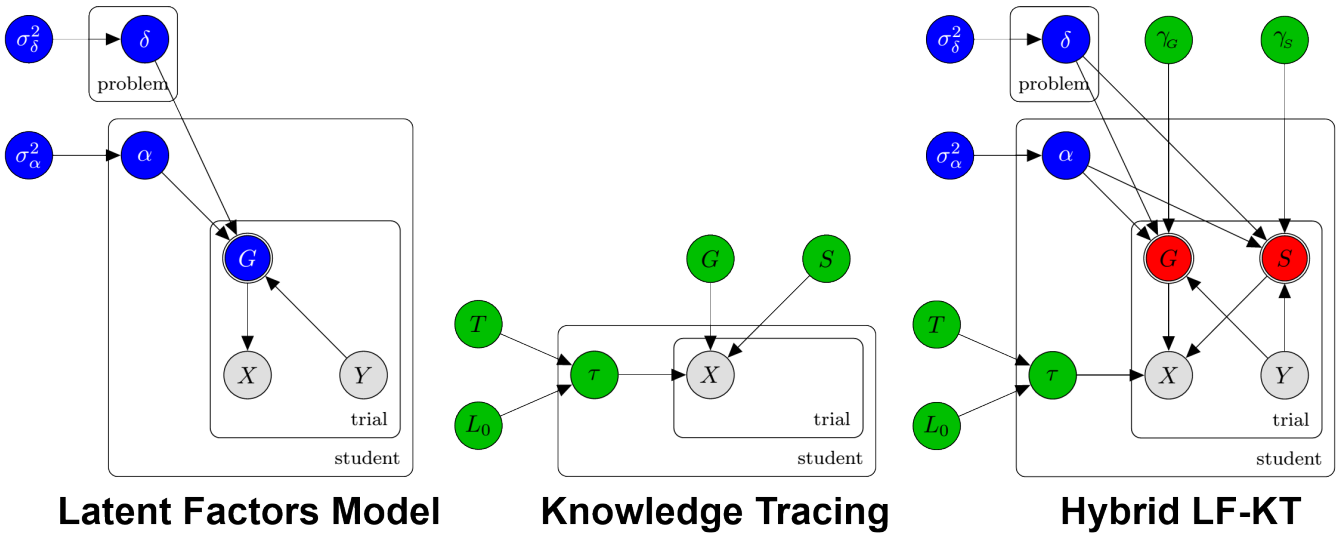

Figure 2: Graphical model depiction of the latent-factor model (left), Bayesian knowledge tracing (middle), and our hybrid LFKT model (right). Following standard notation, shaded nodes are observations, with X denoting the response of a student when problem Y is presented. Double circles denote deterministic nodes. A node's color represents the model that contributed the node with blue, green and red indicating LFM, BKT and LFKT nodes, respectively.

Figure 1 provides an intuition for the value of both student- and problem-specific factors influencing performance. The Figure shows a student × problem matrix, with a cell colored to indicate whether the student solved the problem. As variation in the columns indicate, some problems are more challenging than others. (However, because problem selection and order are partially confounded, one must be cautious in attributing the accuracy effects to intrinsic difficulty of the problem. Regardless of the source of the effect, the presence of the effect is indisputable.)

In the next section, we propose a synthesis of latent-factor and knowledge-tracing models. The synthesis is a natural extension and integration of past work. Indeed, the synthesis is so natural that another paper accepted at EDM 2014 also made this same proposal [12]. We address this highly related work in the discussion section at the end of this paper. A poster at EDM 2013 [26] also explicitly proposed combining latent-factor and knowledge-tracing models. However, their synthesis focused on individuating BKT's initial mastery probability, whereas our effort focuses on individuating guess and slip probabilities.

3. LFKT: A SYNTHESIS OF LATENT-FACTOR AND KNOWLEDGE-TRACING MODELS

In Figure 2, LFM and BKT are depicted in a manner that allows the two models to be superimposed to obtain a synthesis, which we'll refer to as LFKT, depicted in the rightmost panel of the Figure. LFKT personalizes the guess and slip probabilities based on student ability and problem difficulty:

$$\text{logit}(G_{si}|Y_{is} = y) = \alpha_s - \delta_y + \gamma_G \quad \text{and}$$
$$\text{logit}(S_{si}|Y_{is} = y) = \delta_y - \alpha_s + \gamma_S \ .$$

For simplicity, we assume that the effects of ability and difficulty are symmetric on guessing and slipping, though scaling parameters could be incorporated to permit asymmetry. Due to the offsets γ_G and γ_S, we can constrain the expectations $E[\alpha_s] = 0$ and $E[\delta_y] = 0$ with no loss of generality. Specifically, we assume $\alpha_s \sim \mathcal{N}(0, \sigma_\alpha^2)$ and $\delta_y \sim \mathcal{N}(0, \sigma_\delta^2)$, where σ_α^2 and σ_δ^2 are variances drawn from an Inverse-Gamma-distributed conjugate prior.

LFKT can be specialized to the LFM simply by fixing $T = 0$ and $L_0 = 0$. LFKT can be specialized to BKT at the limit of $\sigma_\alpha^2, \sigma_\delta^2 \to 0$.

Table 2: Dataset columns identifying students, problems, skills and correct responses

	Columns
Student	anonymous student ID
Problem	problem hierarchy + problem name + step name
Skill	problem hierarchy + knowledge component
Correct	first attempt

The LFKT model allows for the simultaneous determination of parameters of BKT and LFM. Alternative approaches might include training one model first, freezing its parameters, and then training the other model; or training the two models independently and then using them as an ensemble for prediction. However, simultaneous training allows each component to be informed by the other. Thus, by considering the difficulty of problems, the transition in the knowledge state may become sharper, and by considering the transition in the knowledge state, a better measure of problem difficulty and student ability may be obtained.

4. METHODOLOGY

4.1 Data and prediction task

Our simulation experiments were conducted using three corpora from the PSLC DataShop [17]: *Geometry Area (1996-97)*, from the Geometry Cognitive Tutor [16], *USNA Physics Fall 2006*, from the Andes Tutor [25] and *OLI Engineering Statics Fall 2011* [24]. The Geometry corpus contains 5,104 trials from 59 students on 18 skills, the Physics corpus contains 110,041 trials from 66 students on 652 skills and the Statics corpus contains 189,297 trials from 333 students on 156 skills. Each corpus was divided into skill-specific data sets consisting of the sequence of trials for each student involving problems that require a particular skill. In this paper, we refer to these sequences as student-skill sequences. If multiple skills are associated with a problem, we treat the combination of skills as one unique skill. Trial sequences had mean length 8.0 for Geometry, 4.5 for Physics and 6.0 for Statics.

For reference, Table 2 shows the dataset columns used to identify students, skills, problems and correct responses. The PSLC datashop exports datasets in a common format, which allows us to refer to the same column names for all datasets. The plus sign indicates that the contents of the columns are concatenated together. We attach the problem hierarchy to the skill column following the same practice in [22, 21]. Effectively, this breaks up trial sequences into shorter sequences, which alleviates the problem of students forgetting learned skills over a long time period.

To validate model implementation and parameter settings, we also explored a synthetic dataset obtained by running LFKT in generative mode with the same weak priors used for inference in real datasets. The synthetic dataset contains 50 students and 50 skills. Each skill contains 50 problems and a student may practice a skill for a maximum of 50 trials.

In the literature on student modeling, a variety of measures have been used to evaluate model performance. It seems common to train a model on the entire data set, and to use an AIC- or BIC-penalized measure of fit to estimate performance. We prefer the more conventional approach of partitioning a data set into training and test trials. One way to partition is between the early and late trials in each student's trial sequence. Using this partition, one can predict the future performance for a current student. Another way to partition is by placing some students in the training set and some in the test set. Using this partition, one can predict the performance of the model on previously unseen students. We conduct a separate set of simulation studies for each partitioning.

Model predictions, $\hat{P}$, were evaluated using the log likelihood of the complete test data, i.e.,

$$l = \sum_s \sum_{i=1}^{N_s} \ln \hat{P}(X_{si}|X_{s1}, \ldots, X_{s,i-1}),$$

which can be interepreted as a measure of sequential prediction accuracy for each test trial conditioned on preceding trials in the student-skill sequence.

4.2 Models and implementation

We conducted simulations using the three models in Figure 2—LFM, BKT, and LFKT—in addition to a baseline model. The baseline model gave a fixed prediction equal to the mean response accuracy on each skill in the training set, and was thus independent of trial, problem, and student. To get a better handle on the contribution of student abilities and problem difficulties to model performance, we also tested variants of LFKT that included only abilities or only difficulties. We refer to these variants as BKT+A and BKT+D, where BKT+AD is equivalent to LFKT.

Models containing student ability parameters (LFM, LFKT and BKT+A) were fitted across skills. Thus, a model may use the performance of a student on one skill to infer the student's performance on another skill. This contrasts with most work on modeling with BKT, where models are independently trained on each skill.

LFM and BKT were implemented as special cases of LFKT, in order to use the same code and algorithms for each model. Each model was evaluated in a two-phase process. In the first phase, using the training data ($\{X_{si}\}, \{Y_{si}\}$) and MCMC sampling, a set of posterior samples were obtained on the variables γ_G, γ_S, $\{\delta_y\}$, $\{\alpha_s\}$, L_0, and T. The conditional data likelihood for each student, $P(\mathbf{X}_s|\mathbf{G}_s, \mathbf{S}_s, L_0, T)$ was computed exactly, and therefore sampling of τ was not required. For the other variables, slice sampling was used for a total of 100 iterations after a burn-in of 10 iterations. (These small numbers were sufficient due to the efficiency of slice sampling.) In the second phase, the training samples were used to formulate predictions for the test set. Due to the conjugate prior on the α's, the posterior predictive distribution on test student ability could be determined analytically, i.e., $P(\alpha_{s'}|\{\alpha_s\})$, where s' indexes a test student, and $\{\alpha_s\}$ are the sampled abilities of the training students. A similar predictive distribution could be obtained for $\delta_{y'}$, the difficulty level for a problem y' found in the test set but not the training set, via $P(\delta_{y'}|\{\delta_y\})$.

Weak priors were specified for six variables in the LFKT model: $\gamma_G, \gamma_S \sim \text{Uniform}(-3, 3)$, $L_0, T \sim \text{Uniform}(0, 1)$, $\sigma_\alpha^2, \sigma_\delta^2 \sim \text{Inverse-Gamma}(1, 2)$.

5. EXPERIMENTS

We conducted two experiments to evaluate the models. The first experiment evaluates model performance on the final trials of current students whilst the second evaluates model performance on students held out from training in a particular skill. The two experimental setups are depicted in Figure 3 and are explained in the next two sections.

5.1 Experiment 1: Predicting Performance of Current Students

In this experiment, we ask the question: given the initial responses of a student practicing some skill, how well does the model predict performance on the remaining trials? To answer this question, we grouped trials by skill and student to obtain a list of student-skill sequences. The last 20% of trials from each sequence were placed in the testing set. This design ensures that the models do not have to generalize to new students.

The top row of Figure 4 shows the mean negative log likelihood on the test data. Each graph is for a different data set. Each bar represents the performance score for a given model, with the models arranged left-to-right from simplest to most complex, i.e., from fewest to most free parameters. Smaller scores indicate better performance. The results are consistent across the four data sets: (1) BKT outperforms the baseline model. BKT assumes the student can be in one of two knowledge states, whereas the baseline model assumes a single knowledge state (and a constant probability of correct response across trials for a given skill). (2) When BKT is modulated by latent student ability (BKT+A) or problem difficulty (BKT+D), it outperforms off-the-shelf BKT, with the possible exception of BKT+D in the Geometry data set. (3) LFKT, which incorporates both student abilities and problem difficulties, outperforms BKT as well as the variants that incorporate one latent factor or another. (4) LFKT also outperforms off-the-shelf LFM, indicating that the temporal dynamics of learning incorporated into BKT are helpful for prediction. Thus, we observe clear evidence that the combination of latent factors and knowledge tracing yields a model with greater predictive power than models that have one component or the other.

5.2 Experiment 2: Predicting Performance of New Students

In this experiment, we ask the question: Given a model trained on some students for a given skill, how well does it predict performance of a new student on that skill?

For this experiment, we chose a random subset of students to hold out from each skill. Fifty train/test splits were generated this way using 10 replications of 5-fold cross validation (with an 80%/20% data split). Results were averaged across the 50 test sets.

Test performance in Experiment 2 is shown in the second row of Figure 4, respectively. The pattern of results we observe is identical to that in Experiment 1, indicating that the superiority of LFKT over BKT and LFM does not depend on the specific manner of evaluating the model. (The error bars are somewhat smaller in Experiment 2 than in Experiment 1 due to the fact that the nature of the experiment allowed for more data to be included in the test set.)

We note that Experiment 2 is not purely student stratified because each student had data included in both the training and test sets, albeit for different skills. We conducted a third experiment in which the models were trained not on all skills simultaneously, but on one skill at a time. This training procedure ensures that the models are truly naive to a given student in the test set, which impacts the performance of BKT+A, LFM, and LFKT. Nonetheless, the training data still constrains the student ability distribution, and as a result, the pattern of results still shows LFKT outperforming LFM and BKT.

5.3 Visualization of the Posterior Marginals

One advantage of using a Bayesian modeling approach is that we obtain posterior distributions over model parameters, rather than just point estimates, which allows us to directly quantify model uncertainity about those parameters. In a Bayesian model, we can estimate the joint posterior distribution over the parameters conditioned on the training data. From the joint distribution, which is challenging to visualize, we can compute marginals for each parameter. The marginals are easier to interpret. Because we are using an MCMC sampler, we obtain multiple samples of each parameter setting. The estimated marginal posterior for a parameter is then just histogram of those samples.

To calculate the marginals, we trained LFKT on the entire statics dataset and obtained 1000 samples from the posterior. Figure 5 shows visualizations of the resulting marginal distributions for each parameter. The x-axis in each plot is an index over either students, problems, or skills and each vertical slice of a plot provides the probability distribution over the parameter's value. Probability density is indicated by the color. The targets on the x-axis are sorted by the mean value of the corresponding parameter.

The top two plots in Figure 5 give us a clue about why problem difficulties have a larger effect on prediction performance than student abilities for the Statics data set. The posterior on student abilities are smaller in magnitude than the posterior on problem difficulties. Hence, when abilities are removed (i.e., set to 0) in LFKT to obtain the BKT+D model, the model does not lose much during testing. The model appears to be more certain about student and problem parameters (top row of Figure 5) than skill parameters (the bottom two rows of Figure 5). This difference is reflected in the fact that LFM, which uses the student and problem parameters, outperforms BKT, which uses the skill-specific parameters.

5.4 Execution Time

Even though LFKT combines BKT and LFM, its execution time is longer than the sum of the execution times of the two component models. Under LFM, a modification to a problem's difficulty requires re-evaluating the likelihood of the trials involving that problem. However, modifying a problem's difficulty under LFKT requires re-evaluating the

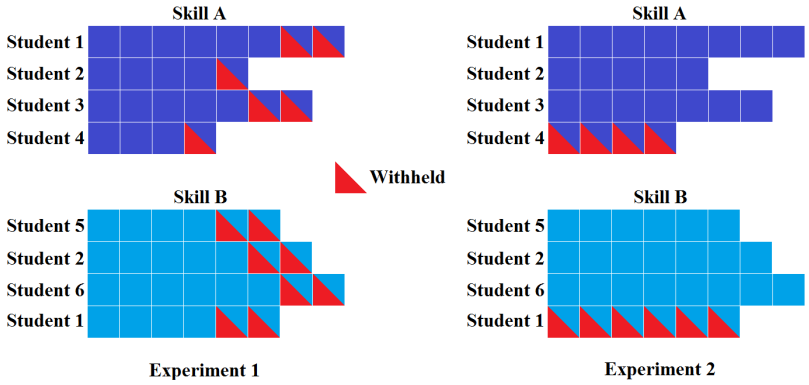

Figure 3: Data split for Experiments 1 and 2 (left and right columns, respectively). The squares represent individual trials and the red triangles represent trials withheld for testing. Squares with different colors belong to different skills.

Table 3: Execution times (seconds)

Dataset	BKT	LFM	LFKT
Synthetic	68.4	108.1	404.0
Geometry	8.0	2.0	14.5
Physics	211.3	412.4	712.4
Statics	175.3	81.0	865.2

likelihood of all the *student-skill sequences* that contain the problem. Table 3 presents the execution time in seconds for each model when training on the entire dataset. The runtime of LFKT is superadditive for all but the Physics data set, which is an anomaly because of (a) the large number of skills which results in short student-skill sequences and (b) the large number of problems which results in a sparse collection of student-skill sequences containing any particular problem. We note that we have made little effort to optimize run times, and alternative approaches (e.g., maximum likelihood parameter estimation) are likely to be significantly faster. Further, run time should not be nearly as important a consideration as model accuracy, so long as run times are tractable, which they clearly are in our simulations.

6. CONCLUSIONS

Within the intelligent tutoring community, there are two common approaches to modeling the performance of a student: Bayesian Knowledge Tracing (BKT) and Latent Factors Models (LFM). BKT is a two state model that attempts to characterize the temporal dynamics of student learning. LFM is a logistic regression model that infers latent factors associated with students, skills, and problems. Two approaches are complimentary, allowing us to synthesize the two into a single model. In this work, we presented LFKT, which integrates BKT and LFM in a mathematically principled manner, and we showed that the synthesis outperforms both BKT and LFM.

We investigated the contribution of individual components and factors within LFKT. Overall, our results indicate that the most important contribution to predicting performance comes from considering problem effects (difficulties), followed by student effects (abilities), followed by skill-specific learning effects (BKT). This ordering holds regardless of whether we are predicting performance on later trials of current students or on complete trial sequences of new students.

One important contribution of the work is the discovery that problem instances drawing on the same skill can systematically vary in difficulty, and inferring the latent difficulty of a problem and incorporating it in a predictive model can significantly bolster prediction accuracy. Although all problems that tax a given skill are equivalent in a formal sense, students are sensitive to the specific instantiation of the skill in a problem. We are aware of three variants of BKT that incorporate this useful fact. The KT-IDEM model [21] incorporates problem difficulties into BKT by fitting separate guess and slip probabilities for each a problem in a skill.

The FAST model [12] provides a general framework for characterizing guess and slip probabilities as a sigmoid function of a weighted linear combination of features. Given student and problem features, FAST discovers weights that are equivalent to the latent ability and difficulty factors in LFKT. However, in FAST, these factors are assumed to be independent for guess and slip probabilities. Thus, both KT-IDEM and FAST have *two* free parameters associated with problem difficulty, whereas LFKT has one one, which is assumed to be symmetric for guess and slip probabilities. This restriction may benefit LFKT in reducing overfitting. Another key difference is that both KT-IDEM and FAST are fit using maxmimum likelihood, whereas LFKT uses MCMC sampling to estimate Bayesian posteriors. The Bayesian approach allows LFKT to generalize to new problems and students in a principled manner. In a recent collaboration with the authors of FAST, we have performed a comparison of LFKT and FAST using the same datasets and evaluation metrics [15].

Another recent development that is complementary to LFKT is a variant of BKT in which the probability of initially knowing a skill (L_0) and the transition probability (T) are individualized to a student [28]. Individualization occurs by

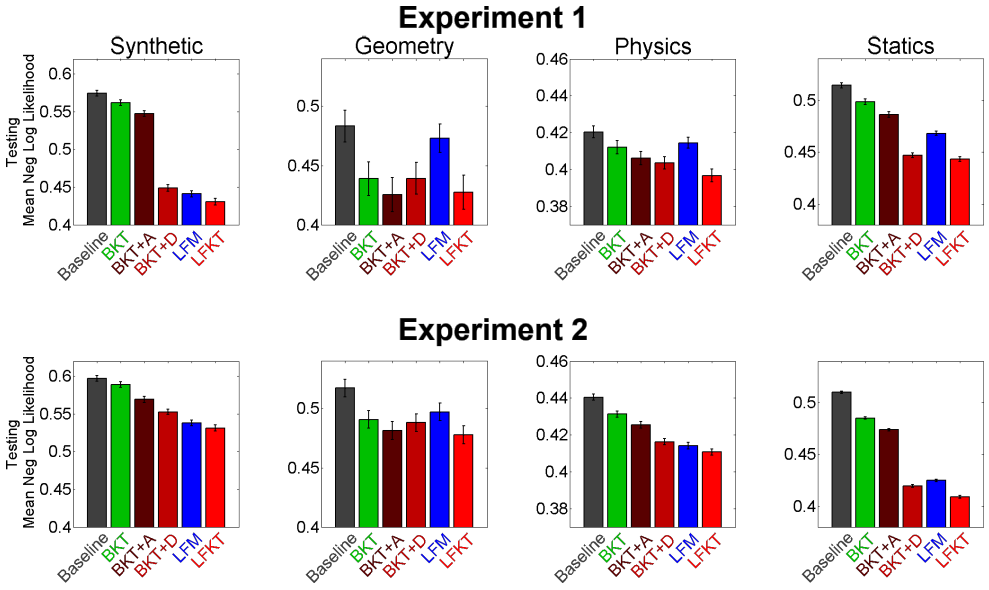

Figure 4: The mean *testing* performance on four data sets (columns) in Experiments 1 and 2 (top and bottom rows, respectively). Each graph shows the negative log likelihood score, averaged across trials, for each of six models. A lower value indicates better performance. BKT+A and BKT+D correspond to LFKT with difficulties set to zero or abilities set to zero, respectively. All trials are weighted equally across skills. Error bars indicate standard errors.

splitting each BKT parameter into skill-specific and student-specific components which are summed and passed through a logistic transform, yielding the BKT parameter value. Although this work mostly parallels ours but focusing on different BKT parameters, our discovery of problem-specific effects makes the intriguing suggestion that one might wish to consider problem difficulty on the transition probability; that is, the probability of learning a skill on a trial may be problem dependendent as well as success dependent.

By understanding the relationship between LFKT and other innovative variants of BKT, we are starting to delineate the space of models of student performance and the critical dimensions along which they vary. This understanding should lead to the emergence of a principled, unified theory that is sensitive to differences among individuals and differences due to the specific content. Such a theory should yield not only improved predictions of student performance but also more effective tutoring systems [18].

Acknowledgments

The research was supported by NSF grants SBE-0542013 and SMA-1041755 and an NSF Graduate Research Fellowship to RL.

7. REFERENCES

[1] R.C. Atkinson. Optimizing the learning of a second-language vocabulary. In *Journal of Experimental Psychology*, volume 96, pages 124–129, 1972.

[2] Ryan S. Baker, Albert T. Corbett, and Vincent Aleven. More accurate student modeling through contextual estimation of slip and guess probabilities in Bayesian knowledge tracing. In *Proceedings of the 9th international conference on Intelligent Tutoring Systems*, pages 406–415, Berlin, Heidelberg, 2008. Springer-Verlag.

[3] Joseph E. Beck. Difficulties in inferring student knowledge from observations (and why you should care). In *Proceedings of AIED2007 Workshop on Educational Data Mining (EDM'07)*, pages 21–30, 2007.

[4] Joseph E. Beck and Kai-min Chang. Identifiability: A fundamental problem of student modeling. In Cristina Conati, Kathleen F. McCoy, and Georgios Paliouras, editors, *User Modeling*, volume 4511 of *Lecture Notes in Computer Science*, pages 137–146. Springer, 2007.

[5] Hao Cen. *Generalized learning factors analysis: improving cognitive models with machine learning*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA, 2009.

[6] Hao Cen, Kenneth R. Koedinger, and Brian Junker. Learning factors analysis - a general method for cognitive model evaluation and improvement. In *Intelligent Tutoring Systems*, pages 164–175, 2006.

[7] Hao Cen, Kenneth R. Koedinger, and Brian Junker. Comparing two IRT models for conjunctive skills. In Beverly Park Woolf, Esma AÃŕmeur, Roger Nkambou, and Susanne P. Lajoie, editors, *Intelligent Tutoring Systems*, volume 5091 of *Lecture Notes in Computer Science*, pages 796–798. Springer, 2008.

[8] Min Chi, Kenneth R. Koedinger, Geoffrey J. Gordon, Pamela W. Jordan, and Kurt VanLehn. Instructional factors analysis: A cognitive model for multiple instructional interventions. In *EDM*, pages 61–70, 2011.

[9] Albert T. Corbett and John R. Anderson. Knowledge tracing: Modelling the acquisition of procedural knowledge. *User Model. User-Adapt. Interact.*, 4(4):253–278, 1995.

[10] Ryan Shaun Joazeiro de Baker, Albert T. Corbett, Sujith M. Gowda, Angela Z. Wagner, Benjamin A. MacLaren, Linda R. Kauffman, Aaron P. Mitchell, and Stephen Giguere. Contextual slip and prediction of student performance after use of an intelligent tutor. In *UMAP*, pages 52–63, 2010.

[11] P. De Boeck and M. Wilson. *Explanatory Item Response Models: a Generalized Linear and Nonlinear Approach*. Springer-Verlag, New York, NY, 2004.

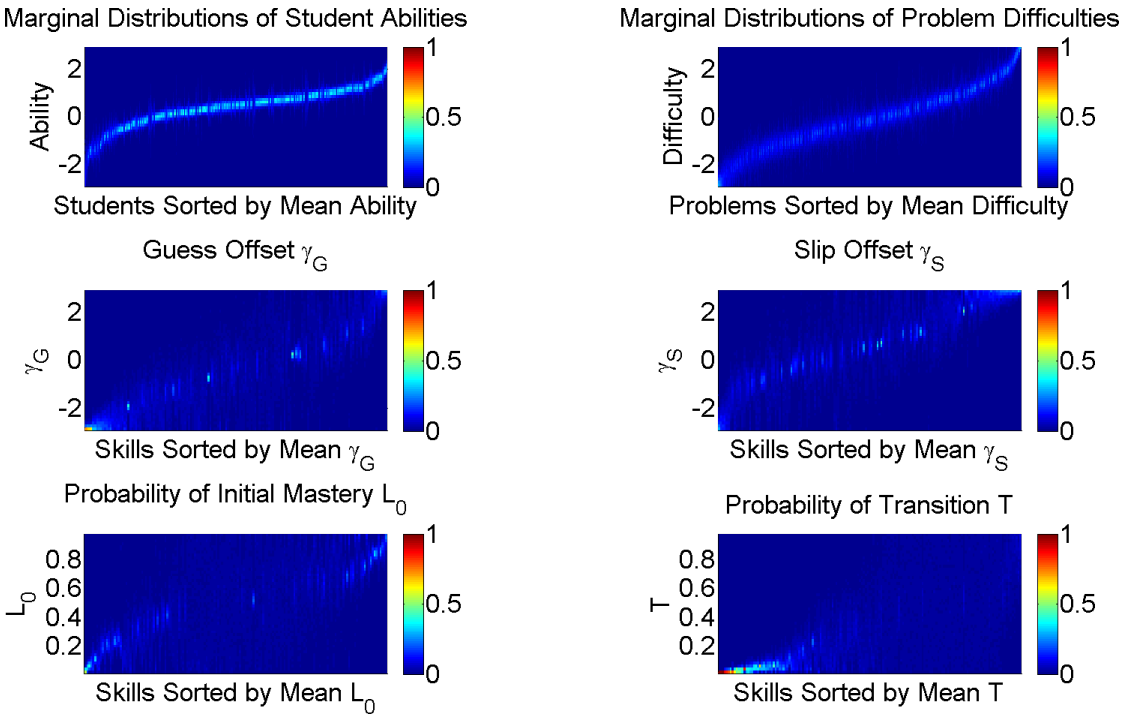

Figure 5: Marginal posterior distributions for each parameter of the model. The x-axis represents the parameter's target (students, problems or skills), the y-axis is the value of the parameter and the color represents the probability of the parameter value for a particular target.

[12] J.P. González-Brenes, Y. Huang, and P. Brusilovsky. General features in knowledge tracing to model multiple subskills, temporal item response theory, and expert knowledge. In *To appear in Proceedings of the Seventh International Conference on Educational Data Mining*, 2014.

[13] Sujith M Gowda and Jonathan P Rowe. Improving models of slipping, guessing, and moment-by-moment learning with estimates of skill difficulty.

[14] KDD cup, 2010. Algebra 1 2005-2006 data set.

[15] Mohammad M Khajah, Yun Huang, José P González-Brenes, Michael C Mozer, and Peter Brusilovsky. Integrating knowledge tracing and item response theory: A tale of two frameworks. In *Personalization Approaches in Learning Environments*, 2014.

[16] Ken Koedinger. Geometry area (1996-97), February 2014.

[17] K.R. Koedinger, R.S.J.d. Baker, K. Cunningham, A. Skogsholm, B. Leber, and J. Stamper. A data respository for the EDM community: The pslc datashop. In C. Romero, S. Ventura, M. Pechenizkiy, and R.S.J.d. Baker, editors, *Handbook of Educational Data Mining*, 2010. http://pslcdatashop.org.

[18] Jung In Lee and Emma Brunskill. The impact of individualizing student models on necessary practice opportunities. In *Educational Data Mining 2012*, pages 118–125. educationaldatamining.org, 2012.

[19] R.V. Lindsey, J.D. Shroyer, H. Pashler, and M.C. Mozer. Improving student's long-term knowledge retention with personalized review. *Psychological Science*, 25:639–47, 2014.

[20] Zachary A. Pardos and Neil T. Heffernan. Modeling individualization in a Bayesian networks implementation of knowledge tracing. In Paul De Bra, Alfred Kobsa, and David N. Chin, editors, *UMAP*, volume 6075 of *Lecture Notes in Computer Science*, pages 255–266. Springer, 2010.

[21] Zachary A Pardos and Neil T Heffernan. KT-IDEM: Introducing item difficulty to the knowledge tracing model. In *User Modeling, Adaption and Personalization*, pages 243–254. Springer, 2011.

[22] Zachary A. Pardos, Shubhendu Trivedi, Neil T. Heffernan, and Gábor N. Sárközy. Clustered knowledge tracing. In Stefano A. Cerri, William J. Clancey, Giorgos Papadourakis, and Kitty Panourgia, editors, *ITS*, volume 7315 of *Lecture Notes in Computer Science*, pages 405–410. Springer, 2012.

[23] Philip I. Pavlik, Hao Cen, and Kenneth R. Koedinger. Performance factors analysis –a new alternative to knowledge tracing. In *Proceedings of the 2009 conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling*, pages 531–538, Amsterdam, The Netherlands, The Netherlands, 2009. IOS Press.

[24] Paul Steif and Norman Bier. Oli engineering statics - fall 2011, February 2014.

[25] Kurt VanLehn. USNA physics fall 2006, February 2014.

[26] Y. Xu and J. Mostow. Using item response theory to refine knowledge tracing. In S. K. D'Mello, R. A. Calvo, and A. Olney, editors, *Proceedings of the Sixth International Conference on Educational Data Mining*, pages 356–7, 2013.

[27] Hsiang-Fu Yu and Others. Feature engineering and classifier ensemble for KDD cup 2010. Technical report, Department of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan, 2010.

[28] Michael V Yudelson, Kenneth R Koedinger, and Geoffrey J Gordon. Individualized Bayesian knowledge tracing models. In *Artificial Intelligence in Education*, pages 171–180. Springer, 2013.