Gong, Y, Beck, J. E., Heffernan, N. T.: How to Construct More Accurate Student Models: Comparing and Optimizing Knowledge Tracing and Performance Factor Analysis. *International Journal of Artificial Intelligence in Education*. Accepted, 2010.

How to Construct More Accurate Student Models: Comparing and Optimizing Knowledge Tracing and Performance Factor Analysis

Yue Gong, *Computer Science Department, Worcester Polytechnic Institute,100 Institute Road, Worcester, MA, 01609, USA, ygong@wpi.edu*
Joseph E. Beck, *Computer Science Department, Worcester Polytechnic Institute,100 Institute Road, Worcester, MA, 01609, USA, josephbeck@wpi.edu*
Neil T. Heffernan, *Computer Science Department, Worcester Polytechnic Institute,100 Institute Road, Worcester, MA, 01609, USA, nth@wpi.edu*

Abstract. Student modeling is a fundamental concept applicable to a variety of intelligent tutoring systems (ITS). However, there is not a lot of practical guidance on how to construct and train such models. This paper compares two approaches for student modeling, Knowledge Tracing (KT) and Performance Factors Analysis (PFA), at predicting individual student trials. We explore the space of design decisions for each approach and find a set of "best practices" for each. In a head to head comparison, we find that PFA has considerably higher predictive accuracy than KT. In addition to being more accurate, we found that PFA's parameter estimates were more plausible. Our best-performing model was a variant of PFA that ignored the tutor's transfer model; that is, it assumed all skills influenced performance on all problems. Two possibilities are that this result is a general one suggesting we consider models that incorporate information from more than the typical handful of skills per problem, or that the transfer model our tutor uses is particularly weak.

Keywords. Student modeling, Knowledge tracing, Performance Factors Analysis, Expectation Maximization, Machine learning, Model fitting approaches, Data aging

INTRODUCTION

Student modeling is one of the major issues for Intelligent Tutoring System (ITS) as it has been widely used for making inferences about the student's latent attributes. Its working mechanism is to take observations of a student's performance (e.g. the correctness of the student response in a practice opportunity) or a student's actions (e.g. the time he stayed for a question), and then use those to estimate the student's underlying hidden attributes, such as knowledge, goals, preferences, and motivational state, etc. Those attributes are unable to be determined directly, thus student modeling techniques have always attracted a great deal of attention.

In ITS, student modeling has two common usages. The first, and most frequently used one, is to predict student behaviors, such as student performance in the next practice opportunity. For example,

Cognitive Tutor, one of the most successful ITS (Koedinger, Anderson, Hadley & Mark, 1997), uses a student modeling technique, knowledge tracing (Corbett & Anderson, 1995), to track student knowledge in order to determine student mastery during problem practicing. Other than using student modeling at run time in an ITS, the technique is also applied to predict student real world performance, Feng, et al. employed item response theory(van der Linden et.al, 1997) and ITS data to estimate student performance in an end-of-year high stakes state test (Feng, Heffernan & Koedinger, 2009). The second usage of student modeling is to obtain plausible and explainable parameter estimates, where plausibility concerns how believable the parameters are, often tested by comparing them to an external gold standard. Being explainable indicates the parameter estimates produced by the model have practical meanings, which can help researchers know more about scientific facts. Beck, et al investigated the helpfulness of helps provided in an ITS by interpreting parameters estimated from learning decomposition (Beck & Mostow, 2008) and an augmented knowledge tracing model (Beck, Chang, Mostow & Corbett, 2008). Other work (Arroyo & Woolf, 2005; Gong, Rai, Beck & Heffernan, 2009) inspected student motivational effects on student learning based upon interpreting their model parameter estimates. Especially for the common issue, student misuse of intelligent tutoring systems (Baker, Corbett, & Koedinger, 2004), the effects of 'gaming the system' and off-task have been examined by using many different student models and understanding the model parameters (Baker & de Carvalho, 2008; Cocea, Hershkovitz, & Baker, 2009; Gong, Beck, Heffernan & Forbes-Summers, 2010). Towards the two common usages, student models are usually evaluated by how well they predict student's behaviors, as well as by parameter plausibility (Beck, 2007; Beck & Chang, 2007).

The importance of student modeling motivates researchers to put more efforts. Lots of work has been done in order to improve student modeling techniques, pursuing higher predictive accuracy and (or) higher parameter plausibility. Baker, et al. (Baker, Corbett & Aleven, 2008) introduced the concept of contextual guess and slip into knowledge tracing to improve the model accuracy. Pardos, et al. integrated individualization into knowledge tracing and showed a reliable improvement in prediction of real world data (Pardos & Heffernan, 2010). Pavlik et al. presented a new alternative student model, Performance Factor Analysis (PFA) and found that it is somewhat superior to knowledge tracing (Pavlik, Cen & Koedinger, 2009). Moreover, there is also a great deal of work focusing their efforts on examining and improving student model parameter plausibility (Beck & Chang, 2007; Gong, Beck & Heffernan, 2010; Pardos & Heffernan, 2010; Rai, Gong & Beck, 2009) as well.

It seems like that the field is making progress, however, we must face a crucial problem: across various models and approaches presented in different work, fair and reasonable evaluations are hard to achieve. Reasons are manifold. Different studies use different data sets, reports different measurement metrics, thus leading to difficulties to compare the models. Even if they did avoid the above differences, it is still likely that they fit their models differently, even though the models they used are the same. Unfortunately, for some models, the way of fitting is not trivial. Knowledge tracing, the model that has been used and researched the most, is one of them. In addition, many models have their own weaknesses or even problems. In this case, the ways of handling the problems could also have impact on model performances, yet researchers don't necessarily deal with them in the same way. Therefore, it is of critical importance to perform comparative analyses of various approaches with taking those issues into account. In this study, we focus our efforts on the following two aspects. 1). compare two competitive student modeling techniques: knowledge tracing (KT) vs. performance factor analysis (PFA). We perform 'within comparisons', where for KT, the comparisons concern the knowledge tracing models fitted by different model fitting approaches and the models with different methods of handling the model's problem (which will be illustrated later); for PFA, a few new

proposed PFA variants are compared, as well as 'between comparisons' in which we give comprehensive comparisons between KT and PFA. 2). Optimize the PFA model. Towards the problems of PFA, we proposed several variants, attempting to improve the model's predictive accuracy.

STUDENT MODELING TECHNIQUES

Knowledge tracing model

There are a variety of student modeling techniques. The knowledge tracing model (Corbett, Anderson, 1995) shown in Fig. 1, has been broadly used. It is based on a 2-state dynamic Bayesian network where student performance is the observed variable and student knowledge is the latent. The model takes student performances and uses them to estimate the student's level of knowledge. After training the model, there are two performance –parameters estimated: slip and guess, which mediate student knowledge and student performance. The guess parameter represents the fact that the student may sometimes generate a correct response in spite of not knowing the correct skill. The slip parameter acknowledges that even students who understand a skill can make an occasional careless mistake. There are also two learning parameters. The first is initial knowledge (K0), the likelihood the student knows the skill when he first uses the tutor. The second is the learning rate, the probability a student will acquire a skill as a result of an opportunity to practice it.

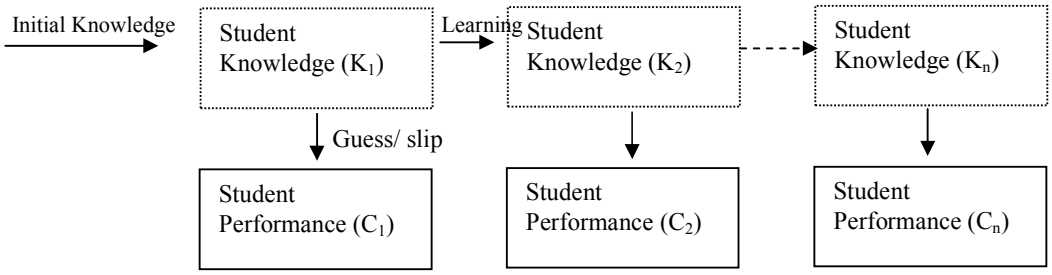

Fig. 1. Knowledge tracing model

When knowledge tracing is used for prediction, the model uses knowledge tracing parameters, usually estimated from the training data, as well as the student's actual performances. The model, using K0 at the very beginning, iteratively updates student knowledge according to the student performance in the previous practice opportunity. The pseudo code used to update the student knowledge in his t^{th} practice opportunity is shown in the following.

```
1    if response_{t-1} == correct
2        k = k_{t-1}*(1-slip)/(k_{t-1}*(1-slip) + guess * (1-k_{t-1}));
3    else
4        k = k_{t-1}*slip/(k_{t-1}*slip + (1-k_{t-1})*(1-guess));
5    k_t = k + (1-k)*learning;
```

Based on the estimated knowledge, the student performance at that particular practice opportunity can be calculated by applying the knowledge tracing equation, shown in Equation 1. $P(correct_t)$ reflects the probability of the student generating a correct response at t^{th} practice opportunity. In this way, the model predicts student performance step by step for every problem solved by the student.

$$P(correct_t) = K_t * (1 - slip) + (1 - K_t) * guess \quad (2)$$

Performance Factor Analysis

Recently, a new alternative student modeling approach was presented by Pavlik et al. (Pavlik, Cen & Koedinger, 2009), Performance Factor Analysis (PFA). PFA is based on reconfiguring Learning Factor Analysis (LFA) (Cen, Koedinger & Junker, 2007).

$$m(i, j \in KCs, n) = \alpha_i + \sum_{j \in KCs} (\beta_j + \gamma_j n_{i,j}) \quad (2)$$

$$p(m) = \frac{1}{1 + e^{-m}} \quad (3)$$

LFA's standard form is shown in Equation 2, where m is a logit value representing the accumulated learning for student i (ability captured by α parameter) using one or more knowledge components (KCs) j. The easiness of these KCs is captured by the β parameters for each KC, and the benefit of frequency of prior practice for each KC is a function of the n of prior observations for student i with KC j (captured by the addition of γ for each observation). Equation 4 is the logistic function used to convert m strength values to predictions of observed probability (Pavlik, Cen & Koedinger, 2009) .

Briefly speaking, PFA takes the form of standard logistic regression model with the student performance as dependent variable. The formula for PFA is very similar, and also uses a logistic model for making predictions. As shown in Equation 4, it reconfigures LFA on its independent variables, by dropping the student variable (α) and replacing the knowledge component variable with the question identity (i.e. one parameter per question).The model estimates a parameter for each item representing the item's difficulty. Thus β parameters no longer capture the easiness of the KCs (also called skills in this paper), but that of the problems. The model also estimates two parameters (γ and ρ) for each skill reflecting the effects of the prior successes and prior failures achieved for that skill. The conversion from the logit value to the prediction of student performance is done by following Equation 3.

$$m(i, j \in KCs, s, f) = \sum_{j \in KCs} (\beta_j + \gamma_i s_{i,j} + \rho_j f_{i,j}) \quad (4)$$

The PFA model can also be viewed as a learning decomposition model (Beck & Mostow, 2008) in that it estimates the different effects of getting a practice opportunity correct or incorrect.

ISSUES WITH KNOWLEDGE TRACING

Model fitting approaches

As pointed out in (Beck & Chang, 2007; Pardos & Heffernan, 2010; Rai, Gong & Beck, 2009), knowledge tracing suffers from two major problems with trying to estimate parameters: local maxima and multiple global maxima. The first one is common to many error surfaces and has known solutions such as multiple restart. The second difficulty is known as identifiability and means that for the same model structure, given the same data, there are multiple (differing) sets of parameter values that fit the data equally well. Based on statistical methods, there is no way to differentiate which set of parameters is preferable to the others. Different approaches have various criteria to fit the data, and thus produce different parameter estimates and further lead to different predictive abilities. Therefore, we explored the impact of modeling fitting approach on model accuracy and plausibility.

The Expectation Maximization (EM) algorithm (Moon, 1996) is a model fitting approach for KT. It finds a set of parameters that maximize the data likelihood (i.e. the probability of observing the student performance data). EM processes student performance as a piece of evidence with time order, and uses this evidence for the expectation step where the expected likelihood is calculated. The model then computes the parameters which maximize that expected likelihood. The algorithm iteratively runs these two steps until it finds the final best fitting parameters. There is no guarantee of finding a global, rather than a local, maxima.

Recently, the brute force approach has been proposed for estimating parameters for KT (Baker, Corbett & Aleven, 2008). Contrary to EM that maximizes the data likelihood, it attempts to minimize the sum of squared error (SSE). Originally, KT's parameters are continuous, so that there is no way to compose a finite search space, which, however, is a must for an exhaustive search. We used the source code provided by Ryan Baker, which resolves the issue by only considering two decimal places of precision. In this way, the parameter space is reduced from infinity to 99^4 possible parameter sets for each skill (i.e. there are four parameters for each skill and each of them has 99 possible values ranging from 0.01 to 0.99). Initially, every parameter starts from the value of 0.01 and is incremented by 0.01 on every iteration. Ultimately, for each skill, it finds the lowest SSE, and the corresponding set of parameters for that skill.

The major drawback is that the method suffers from a severe computational cost problem due to the large search space, so most of the time the search space is cut down even smaller by setting searching boundaries. In this study, we applied the brute force approach on the knowledge tracing model with two different settings. The first one is to explore the entire parameter space. Namely, in the search procedure, no boundaries are designated. The search algorithm is free to visit every combination of the parameters. The second setting is to set search boundaries. Original work (Pavlik, Cen & Koedinger, 2009) compared PFA and KT fitted with brute force. In our study, a comparison process with minimal differences is preferred. Therefore, in order to make a careful comparison, we followed the same protocol as the original work followed. Specifically, we used the same set of bounded ceiling values for the four parameters, so that the maximum probabilities of initial knowledge, guess, slip and learning are 0.85, 0.3, 0.1 and 0.3, respectively.

Conjugate Gradient Descent, an optimization method used to solve systems of equations, is used to estimate parameters in the CMU cognitive tutors. Chang et al. (Chang, Beck, Mostow & Corbett, 2006) found that EM produces models with higher predictive accuracy than Conjugate Gradient Descent.

Unlike the KT model, the family of logistic learning decomposition models is based on the form of standard logistic regression, so that the model fitting procedure is ensured to reach global maxima; thus resulting in unique best fitting parameters. Consequently, for PFA, the model fitting approach is not an issue.

Problem with multiple skill questions

The KT framework has a major weakness: when there is more than one skill involved in a question (called a multi-skill question), the model lacks the ability to handle all the skills simultaneously, because KT works by looking at historical observations on *a* skill. However, in some tutors, a question is usually designed to require multiple skills to achieve a correct answer. If a student model cannot accommodate this common phenomenon well, its ability of making plausible parameter estimates and accurate prediction is likely to be weakened.

A common solution is to associate the performance on the multi-skill question with all requested skills, by listing the performance repeatedly, so that the model sees this piece of evidence multiple

times each of which is for one of the requested skills. (e.g., Pardos, Beck, Ruiz & Heffernan, 2008). Thus, when we train a KT model, a multiple skill question is split into multiple single skill questions.

This strategy enables parameter estimation to proceed, but increases the probability of overfitting and also results in an accompanying problem: multiple predictions. Each split performance is associated with a particular skill that has its own set of parameters, which are used to calculate the student's predicted performances. It is highly likely that those predicted values are not equivalent, which means for the same student, on the same practice opportunity, our models make different claims about how likely he is to produce a correct response. Given the conflicting predictions, some means of making a prediction is needed.

In this study, we attempted two approaches to address the problem. The first is similar to (Pardos, Beck, Ruiz & Heffernan, 2008) and inspired by the joint probability in Probability Theory. The probability a student generates a correct answer in a multi-skill question is dependent on his ability to achieve correctness in all required skills. Therefore, we multiplied each skill's predicted performance together and assign the product as the new predicted performance for all corresponding observations. Yet, the reasonableness of this method relies on an assumption: how likely a student can answer correctly for one skill must be independent of the probability that he responds correctly in another skill.

The second approach, on the other hand, takes the minimum probability of the predicted performances as the final value. The intuition behind this strategy is the likelihood of a student gets a correct answer is dominated by his knowledge of his weakest skill.

The above strategies are necessary options for KT due to its lack of ability to handle multi-skill performances. However, it is not the case for PFA, as it has the ability. Therefore in this study, in addition to repeating multiple skill questions like we do for KT, we also examined PFA when it works in its natural spirit where we fit the model with the data that still keep the original multi-skill performances.

ISSUES WITH PFA

The Performance Factor Analysis model learns the parameters of item difficulty, the effect of prior successes and the effect of prior failures for each skill. When the model counts the numbers of prior successful and failed practices, it doesn't consider the order of those practices occurred, which might be a potential problem. Consider the following example, a student is about to solve a question of Pythagorean Theorem. Before that he has done four questions involving the same skill. Suppose at present, the model counts that there were two correct answers and two incorrect answers in the past, we are not able to retrieve his historical records, as having the numbers of correct and incorrect answers doesn't suffice to reconstruct the order of those answers . There are six possible conditions all of which could lead to the current counts. It is reasonable that we hypothesize that the effects on the current practice from previous problem solving might be different if the order of correct responses varies. For example, consider the case of a student getting the first two correct and the last two incorrect, as compared to a student whose performance was reversed. Therefore, we attempt to address this concern with PFA.

Our main idea is to employ data aging to take into account the performance order. Based on the assumption: the further back the past practice was, the less it impacts the current question, we introduced a decay factor, δ ($0 < \delta \leq 1$), that updates the counts by decreasing the importance of prior performances. The formulas are as follows:

$$success\_count_t = \sum_{1 \le k \le t-1} P_k * \delta^{(t-1-k)} \quad (5)$$

$$failure\_count_t = \sum_{1 \le k \le t-1} |P_k - 1| * \delta^{(t-1-k)} \quad (6)$$

These two formulas replace $s_{i,j}$ and $f_{i,j}$ in the original PFA formula, where in formulation, $s_{i,j}$ and $f_{i,j}$ are represented as:

$$s_{i,j} = \sum_{1 \le k \le t-1} P_k \quad (7)$$

$$f_{i,j} = \sum_{1 \le k \le t-1} |P_k - 1| \quad (8)$$

Same as the standard PFA, the new count function only considers questions on the same skills. In the Equations, t indicates that the current question is the t^{th} question that the student is about to solve. P_k is the correctness in the k^{th} practice opportunity. δ is the decay factor. Take the above example, suppose the student has done four questions of Pythagorean Theorem, the sequence of performances were correct, correct, incorrect and incorrect (1,1,0,0), and the decay factor is 0.9 . According to our formulas, the number of prior successes would be the sum of $1*0.9^3+1*0.9^2+0*0.9^1+0*0.9^0$, which equals to 1.5, while the number of prior failures would be the sum of $0*0.9^3+0*0.9^2+1*0.9^1+1*0.9^0$, which equals to 1.9. Contrariwise, a student who got the first two items right and the last two items wrong would have a correct count of 1.9 and an incorrect count of 1.5. In this way, the model is able to differentiate the performance orders.

By giving more weights to the more recent practices, those practices' effects are updated as more important than ones further in the past. This new approach has the benefit of supporting various assumptions in terms of the decay impact of previous practices. Smaller δ implies that the further practices rapidly become less important. On the contrary, when δ is 1, we have the classic PFA. In this study, we chose δ=0.9 as we don't want to eliminate the effects of further practices too quickly.

We use the formulation is Equations 5 and 6 for expositional clarity. However, it is possible to compute the aging values without maintaining the student history on the skill, and instead at each time step simply multiply the success and failure counts by δ.

METHODS

For this study, we used data from ASSISTment (Razzaq et. Al, 2005), a web-based math tutoring system. The data are from 343 twelve- through fourteen- year old 8th grade students in urban school districts of the Northeast United States. They were from four classes. These data consisted of 193,259 problems completed in ASSISTment during Nov. 2008 to Feb. 2009. Performance records of each student were logged across time slices for 104 skills (e.g. area of polygons, Venn diagram, division, etc).

Previous work compared KT and PFA models, and found PFA to be superior (Pavlik, Cen & Koedinger, 2009). In this study, we ran replication study, but also focusing on examining design decisions of each model. For KT, we investigated impacts when using different model fitting approaches. We also attempted and tested different approaches to handling multiple-skill problems. For PFA, we presented several PFA variants in order to optimize the model performance. In addition, we examined two conditions of the PFA model when it keeps negative learning rates and doesn't. In PFA, the impact of a correct response for a skill is referred as the learning rate of that skill. In the original work, the authors restrict the learning rate to be non negative. Our goal here is to see whether

and how negative learning rates impact the model's predictive accuracy. We present our comparison results based on both predictive accuracy and parameter plausibility.

We used Bayesian Network Toolkit for Student Modeling (BNT-SM) (Chang, Beck, Mostow & Corbett, 2006) to perform the EM algorithm for the knowledge tracing model to estimate the model parameters. We used Ryan Baker's unpublished java code to accomplish the brute force model fitting method on KT. We also replicated the PFA model using the same model settings as in Pavlik, Cen & Koedinger, 2009, except where noted below.

We fit the three models with the pre-processed data in terms of converting multiple skill questions into multiple single skill questions. For PFA, we also examined it by fitting the original data which keeps multi-skill questions as a single unit. We refer to the PFA model handling multi-skill performances as PFA-Multi and the other dealing with multiple single questions as PFA-Single. We did 4-fold crossvalidation at the level of students, and tested our models on unseen students, which is different from what Pavlik, et al. did (Pavlik, Cen & Koedinger, 2009). They conducted 7-fold crossvalidation and tested their models on seen students' unseen performances. We prefer to hold out at the student level since that results in a more independent test set.

In the next section, we report the comparative results by listing the mean values of measurements across four folds. Also we used cohen's d to indicate effect sizes and report them in parentheses in the tables, except where noted. To evaluate the models, we also perform statistical tests. All statistical tests we used, except where noted, are paired two-tailed t-tests using the results from the crossvalidation with degrees of freedom of N-1, where N is the number of folds.

RESULTS: PREDICTIVE ACCURACY

Predictive accuracy is the measure of how well the instantiated model fits the test data. We used two metrics to examine the model predictive performance on the unseen data set: R^2 and AUC (Area Under Curve) of ROC curve (Receiver Operating Characteristic curve). R^2 is a measure of how well a model performs in terms of accurately predicting values for each test data point, where the baseline it compares to is predicting the data points by just guessing the mean of the sample. AUC of ROC curve evaluates the model's performance on classifying the target variable which has two categories. In our case, it measures the model's ability to differentiate students' positive and negative responses.

One crucial point to keep in mind in examining these results is the R^2 values are for predicting individual student trials. That is, for each student response our models make a prediction and are scored on their accuracy. We compute error = (predicted value – actual value)^2, and compute R^2 as:

$$R^2 = 1 - \frac{\sum (predicted\_value - actual\_value)^2}{\sum (mean - actual\_value)^2}$$

Note that for predicting individual trials, R^2 values are typically fairly low (Heathcote, Brown & Mewhort, 2002; e.g. Beck & Mostow, 2008). If instead we predict aggregate performance and plot learning curves as are frequently seen in talks and papers, we have an R^2 of 0.88; so our skill model is reasonably accurate and our data register student learning. Our position is that the lower R^2 is not a sign that there is a problem with our models; rather it is what results when one closely examines the factors that influence student performance.

Knowledge tracing: using different model fitting procedures

In the first part of this section, we compare predictive accuracy of the knowledge tracing models fitted by the brute force algorithms of two different settings. The first model, where the brute force was restricted by the boundaries, is labelled as BF-Restricted. The other model, which explored the entire search space, is labelled as BF-Full. We also reported the size of the search spaces.

The first two columns of Table 1 show the results of the comparisons for the two metrics. The values are calculated by averaging corresponding numbers obtained in the 4-fold crossvalidation. The numbers in the parentheses indicate the effect sizes of the comparisons from cohen's d.

Since R^2 measures a model's predictive ability by comparing to a naïve model, which guesses the mean as the prediction value for each individual data point, 0 indicates that the model has no prediction power once knowing the mean value of the target to be predicted; negative values suggest that the model is even worse than just guessing the mean in every prediction. For the other metric, AUC of 0.5 is the baseline, which suggests random prediction: there is no relationship between the predicted value and the true value. Our results suggest that the size of the parameter space considered by the algorithm does matter. Based on a paired two-tailed t-test using the results from the crossvalidation, we found that BF-Full is able to outperform its counterpart and also significantly so in both metrics ($p<0.05$). The effect size of R^2 is large, while even in AUC, the effect size is medium. However, one thing worth pointing out is that the improvement is less exciting if we take into account the cost it needs. For BF-Full, the search space is 130+ times as big as used by the restricted brute force, considering the baseline is already quite large (765,000 data points), BF-Full needs much more resources and computational time. For our dataset, over five days were required for each fold of the cross-validation.

Table 1. Crossvalidated predictive accuracy comparison between two KT models with brute force of two settings

	R^2	AUC	size of search space
BF-Restricted	0.036	0.656	$7.65x10^5$
BF-Full	0.072 (2.49)	0.662 (0.45)	$9.6x10^7$

Other model fitting procedures than brute force can be used to estimate parameters. Table 2 compares the models' predictive accuracy when applying brute force and expectation maximization (see default models). Although most numbers seem very close, EM outperforms BF-Restricted in both metrics ($p<0.01$ in R^2; $p=0.02$ in AUC), and is nearly identical with BF-Full. The effect size of EM against BF-Restricted (for a clear representation, we didn't list it in the table) is relatively smaller than that of BF-Full: 2.44 for $R^2$and 0.38 for AUC. In the aspect of computational time, although EM needs to run several iterations until convergence occurs, the time consumed is at the same level as BF-Restricted.

Table 2 Crossvalidated comparisons of the default models and the models solving multi-skill questions

		R^2	AUC
BF-Restricted	Default	0.036	0.656
	Multiplication	-0.144	0.656
	Min	0.046 (0.72)	0.670 (1.2)
BF-Full	Default	0.072	0.662
	Multiplication	-0.191	0.634
	Min	0.073 (0.15)	0.677 (1.36)
EM	Default	0.072	0.661
	Multiplication	-0.175	0.633
	Min	0.073 (0.14)	0.676 (1.28)

Knowledge tracing: handling multi-skill questions

Given the problem of multi-skill questions in KT, we compared the two proposed approaches for predicting performance (multiplication and min()) with the default models, which have no designed techniques to handle multi-skill questions. Those default models, after splitting multiple skill questions into multiple single skill questions, would make multiple, different predictions, on student performance on the question, and one for each skill.

We found the approach of calculating the product results in worse predictive accuracy for all model-fitting approaches. In a sense, this result means the skills are not truly independent of each other. In contrast, taking the minimum value of the predicted performances provides more accurate models. As shown in Table 2, comparisons between mean values suggest that the min models are generally better than the default models. The AUC values are reliably different ($p<0.05$) in every pair of the comparisons, whereas in R^2 we failed to find any reliable differences. Another consistent finding lies in the effect sizes (shown in parentheses). The model of taking the minimum value achieved a large effect size in AUC, but not so much in R^2. This indicates that this approach helps produce more accurate classification in a great amount, yet not so in terms of minimizing the prediction errors, especially for BF-Full and EM, which are the two models that have explored the search space completely.

PFA: bounding learning rates

One problem of PFA models is they could produce negative learning rates due to over fitting or due to sampling bias in the data, so in the original work (Pavlik, Cen & Koedinger, 2009), the researchers set 0 as the lower bound for the impact of getting a question correct. In order to understand whether and how much the learning rates would impact the model's performance, we examined the models where the negative learning rates were maintained, as well as the models with manual intervention where the learning rates are bounded as non negative in the training procedures.

We tested the impact of allowing negative learning rates for all of our variants of the PFA algorithm (except the All Skill model, described below). We found that in all versions of PFA, the models produced a small number of skills with negative learning rates, which varied from 0 to 5 among 104 skills. Moreover, the impact of the negative learning rates is also very small. For most of the 12 pairs of comparisons (three versions of PFA and four folds for each version), the R^2 values and

AUC values are almost identical. Only in two comparisons, R^2 values vary since the fourth decimal and one pair of AUC values don't agree with each other from the third decimal. Take the comparison between the PFA model and the PFA model with bounded learning rate as example, the averaged R^2 values of the two models are 0.167**5**12 and 0.167**6**46; the AUC values of the models are completely the same, which is 0.74475. Most comparisons have this phenomena, thus we don't report the comparative results in a table for less redundancy.

In our prior work (Gong, Beck & Heffernan, 2010), we used a computationally expedient approach, which may have been flawed and diverged slightly from the approach used in (Pavlik, Cen & Koedinger, 2009), of preventing learning rates from being negative. For those learning skills whose learning rates were less than 0, we replaced the learning rate with a 0. This approach differs from (Pavlik, Cen & Koedinger, 2009), as they forced the parameter to be non-negative, which potentially results in a slightly different model-fitting solution. Therefore in this study another goal was to investigate how much the results would vary when using the correct approach. The results from this study verified that the approach we used in our prior work (Gong, Beck & Heffernan, 2010) is an acceptable alternative (at least for our dataset where the automatically machine learnt models yielded few skills with negative learning rates.) as we found the differences in the R^2 values of the two approaches are tiny: the differences start from the fourth decimal.

PFA: handling multi-skill questions

Different from the knowledge tracing model, the PFA model has the inherent ability to handle problems that require multiple skills. In Pavlik et al.'s work, when comparing KT and PFA, the researchers gave PFA the dataset organized in the same way as for KT (Pavlik, Cen & Koedinger, 2009). Therefore, the PFA model also treats a multiple-skill question as multiple single-skill questions. In our study, first of all, we wanted to make a good replication with minimal differences from the previous work (Pavlik, Cen & Koedinger, 2009), so we also trained and tested PFA (PFA-Single) by giving it the identical datasets as KT used. Secondly, it is important to know how PFA performs in its natural way: handling multi-skill questions directly. Hence, for this purpose, to make a fair comparison, we also fitted and tested the PFA model (PFA-Multi) using the same datasets as we used for KT, but all multiple skill questions were maintained, i.e. all KT and PFA models were fitted by the same data which however were organized in different ways..

Table 3 Crossvalidated comparisons between the PFA models handling multi-skill questions

	R^2	AUC
PFA-Single	0.151	0.732
PFA-Multi	0.168 (2.27)	0.745 (2.19)

As seen in Table 3, PFA-Multi results in stronger performances in both metrics, and the differences are statistically reliable at p values<0.01. The effect size, cohen's d, is large. This implies that the ability of PFA to handle multiple skill questions directly indeed benefits the prediction; therefore, PFA should be used in its original spirit as it would result in better predictive performance.

PFA: addressing the issue of performance order

One potential problem of the PFA model is that it ignores the order of practice opportunities, i.e. the model treats all observations equally important regardless of when they happened. We are interested in understanding whether there is impact from considering the order of the observations. We introduced decay factor to solve this issue. A sequence of prior practices with its own correctness order would be

mapped to a unique count of prior successes and a unique count of prior failures. With this new feature, the PFA model is able to predict the student's next performance not only based on the number of previous responses, but also the order of the responses occurred.

Table 4. Crossvalidated comparisons between the classic PFA and a variant PFA that handles the problem of performance order.

	R^2	AUC
PFA-Classic	0.168	0.745
PFA-Decay (δ=0.9)	0.172 (0.51)	0.748 (0.45)

Table 4 shows the comparison between the classic PFA and the version of PFA dealing with the problem of performance order. In both metrics, the PFA with decay factor, δ=0.9, produces higher predictive accuracy than the classic PFA and also reliably so supported by paired two-tailed t tests with p values less than 0.001. We noticed that although our proposed approach goes towards the right direction, as it results in better performance, the effect size is not large, suggesting that more work is needed here.

PFA: ignoring the transfer model

The classic PFA model uses item difficulty and effects from prior practices to predict an incoming question. First it needs to identify which skills are used in this question, and then looks at the prior practices which are associated with those skills. This approach ignores the potential impact of other skills and only considers skills tagged by our subject-matter expert as being relevant to the question. This assumption is reasonable and easily understandable, as only the student's proficiencies on those relevant skills would contribute the question solving. In this study, however, we also presented a new variant of PFA which seems opposite to that assumption. Rather than just taking the skills involved in a question; the model is trained and tested by considering all the skills in the dataset. The intuition is that there might be relationships that are not well captured by the transfer model. For example, perhaps overall student competency is an important predictor? Or perhaps skills involve a broader range of skills than our subject matter expert believed. This model assumes that the probability a student successfully solves a problem might also depend on his proficiencies on other skills. However, there are no easy ways to identify which other skills are important to a given skill, therefore in this study, we used all skills.

We compared the proposed PFA variant with the classic PFA. As seen in

Table 5, the PFA-All skills model seems to beat the classic PFA in both metrics in a great amount. However, we failed to find any reliable differences between these two models, even though the mean values appear a trend suggesting PFA-All skills is probably better than the classic PFA and the cohen'd values suggest that the effect size is large.

One reason for the lack of reliable difference is the relatively low statistical power of the t-tests since we only have four independent observations (one for each fold of the cross validation). Therefore, we designed a special t-test for this comparison. We conducted a paired two tailed t-test between the R^2 values of the data points of each student. In each fold of the cross validation, we calculated R^2 treating students as units. Specifically, a student's performances were grouped together and treated as a subset of the test data. For each subset, we computed the R^2 value. As a result, we had N R^2 values, where N is the number of students that were in the test dataset. These N R^2 values were the independent observations in the t-test. We did four t-tests in total, and one for each fold. In all four comparisons, the mean R^2 values of PFA-All skills are superior and p values from the four t-tests vary from 0.005 to 3.81×10^{-32}. This statistical test indicates that PFA-All skills model has better

performance in terms of predictive accuracy. Interpreting this result is complicated, and more work, using transfer models developed by subject matter experts and for other domains, will help considerably in better understanding it. At present we view the approach as an intriguing option.

Table 5. Crossvalidated comparisons between the classic PFA and all-skill PFA

	R^2	AUC
PFA-Classic	0.168	0.745
PFA-All skills	0.181 (0.89)	0.756 (1.04)

Overall comparison: predictive accuracy across KT and PFA

In this section, we compared four main models. The knowledge tracing model fitted by brute force with restricted search area and the PFA model taking multiple single questions have been compared by other researchers (Pavlik, Cen & Koedinger, 2009). Although the above two models are not the best, we still provide the comparisons of them, because in this study we replicated the comparisons using the same model settings as used in (Pavlik, Cen & Koedinger, 2009) so as to understand the models' performances across different studies.

Other than these two models, we also examined the knowledge tracing model fitted by the expectation maximization algorithm, as we want to conduct a direct comparison between the PFA model and the KT model with the alternative model fitting procedure. In addition, we compared the above three models with the best variant of PFA where it considers all skills. In our results, we also reported the number of parameters produced by each model. In this section, we didn't list effect sizes in the table as most of them have been presented before.

Table 6. Crossvalidated comparisons among main models

	R^2	AUC	# of parameters
KT + EM	0.072	0.661	416
KT + BF-Restricted	0.036	0.656	416
PFA-Single	0.151	0.732	1013
PFA-All skills	0.181	0.756	1013

Divergent with the finding of Pavlik et al. (Pavlik, Cen & Koedinger, 2009), which was that the PFA model is somewhat superior to the KT model, the PFA model, at least on our dataset, outperforms the knowledge tracing model in a great amount regardless of model fitting procedures used for KT. Since we fit both the PFA and KT models to the same dataset where multiple skill questions were split into multiple single skill questions, in this way, a fair comparison is believed to perform, as such dataset favours neither model. We have already shown that PFA-Single is the weakest among all PFA variants, so the comparison in Table 6 suggests that the other PFA variants would be much stronger than the KT model. In the last row of Table 6, we see that compared to KT +EM, the R^2 value is improved 150% by the PFA-All skills model. The effect sizes in both metrics are also large: 6.4 for R^2 and 7.07 for AUC.

We noticed that PFA produced more parameters than KT, which seems inconsistent with the results reported in (Pavlik, Cen & Koedinger, 2009).But, given that the number of parameters in PFA = 2*# of skills + # of items, while the number of parameters in KT is 4*# of skills. Consequently having fewer items favors PFA, while fewer skills benefits KT with respect to yielding fewer parameters.

Although we have reported AUC values for each model, which are used to evaluate the models' classification performances, it is important for researchers to understand which models are better and in which conditions they are. We used confusion matrices to visually understand the models' misclassifications. We compared four models in the confusion matrices. For KT+BF, we used the model that has no restrictions on the search space. We chose 0.5 as the cut-off point. Specifically, for an instance, if the model produces the predicted value which is less than 0.5, we labeled those instances as incorrect answers; otherwise, 1 would be assigned indicating a correct response.

Table 7 provides a deeper examination about the performances of the best-performing models for KT and PFA. For KT, we used both EM and the BF-Full. For PFA, we used the decay model to account for performance ordering and the PFA variant that ignored the transfer model, since those versions performed the best.

Table 7 Confusion matrices

Predicted / Actual	KT + EM		KT+BF-Full		PFA-Decay		PFA-All Skills	
	0	1	0	1	0	1	0	1
0	0.11	0.26	0.11	0.26	0.16	0.20	0.16	0.19
1	0.08	0.55	0.09	0.55	0.09	0.55	0.09	0.56

In Table 7, the four cells in each model's unit represent the percentages of instances that fall into the categories. We showed that the family of the PFA models outperforms the KT models in a great amount in the previous sections. The confusion matrices expose where the power comes from. Compared the two KT models and the two PFA models, we see the big difference lies in false positive which is corresponding to the right up corner in the unit of each model. It indicates that the model claims the students would produce correct responses, while it is an incorrect prediction. We found the PFA models perform much better in this case, although with substantial room for improvement. Moreover, we noticed that generally all models perform better when the students' responses are correct. Among all correctly answered questions, 86%-87% instances are correctly predicted by the four models, while the number is 30% for KT and averagely 45% for PFA when dealing with the instances with incorrect responses. This finding suggests that even given the best model we have so far, still the classification accuracy for incorrect responses is less than 50% which seems disappointed, but the up side is that the numbers guide our research direction towards the efforts of reducing false positive in the student models.

RESULTS: PARAMETER PLAUSIBILITY

Predictive accuracy is a desired property, but ITS researchers are also interested in interpreting models to make scientific claims. Therefore, we prefer models with more plausible parameters when we want to use those for scientific study. However, this is non-trivial due to the lack of gold standards to quantify the parameter goodness. In our study, we used the two metrics we explored in (Rai, Gong & Beck, 2009).

Broken learning rates

For the first metric, we inspected the number of practice opportunities required to master each skill in the domain. We assume that skills in the curriculum are designed to neither be so easy to be (on average) mastered in three or fewer opportunities nor too hard as to take more than 50 opportunities. We computed the number of opportunities needed to master a skill by treating student performance as

unobserved, and used the learning rate to update the probability of knowledge. We define mastery as the same way as was done for the mastery learning criterion in the LISP tutor (Corbett,2001): students have mastered a skill if their estimated knowledge is greater than 0.95. Based on students' prior knowledge and learning parameters, we calculated the number of practice opportunities required until the predicted knowledge exceeds 0.95. Then, we compared the number of skills with unreliable values in both cases (fewer than 3 and more than 50).

Table 8 shows the average numbers of skills calculated across 4-fold crossvalidation and the corresponding standard deviations in the parentheses. The results are consistent in the two conditions. BF-Restricted results in 0 skills with too fast mastery rate and the fewest skills mastered too slowly. It is reasonable, as one of the reasons of assigning boundaries to this model fitting approach for KT is to prevent the model from producing unbelievable parameters. We also see the support from the opposite side where when brute force is released from the bounded search area, the model, BF-Full, generated the most extreme cases in both conditions. While EM is in the middle, contrast to BF-Restricted, the model fitting procedure is allowed to explore the entire search surface and no manually assigned boundaries are needed.

Table 8. Comparison of extreme number of practice until mastery (fractional cases are due to taking the mean across four folds)

	# of skills with # of practices >=50	# of skills with # of practices <=3
BF-Restricted	19 (2.94)	0 (0)
BF-Full	36.5 (2.63)	4.25 (0.96)
EM	28 (1.89)	1.5 (0.58)

Student parameter accuracy

The second metric is using external measurement to evaluate parameter plausibility.

The students in our study had taken a 33-item algebra pre-test before using ASSISTments. Taking the pre-test as external measure of incoming knowledge, we calculated the correlation between the students' initial knowledge estimated by the models and their pretest scores.

In other to acquire student's initial knowledge parameter, we used KT to model the students instead of skills (see Rai, Gong & Beck, 2009 for details). Since PFA has no student parameter by default, we tweaked it to include student as an additional independent variable. In Table 9, we see that the PFA model fitted by the data keeping multi-skill performances produces the strongest correlation. KT+BF surprisingly shows a higher ability to estimate plausible parameters than KT+EM. The KT+EM is reliably different ($P<0.05$) PFA-Multi; none of the other differences is reliable.

Table 9. Comparison of parameter plausibility: correlations between model estimates and external standards.

	KT+BF-Restricted	KT+EM	PFA-Multi
Correlation	0.865	0.827	0.895

CONTRIBUTIONS, FUTURE WORK AND CONCLUSIONS

This paper extends the literature on evaluating student modeling techniques and comparing techniques against each other. As data sets become larger and we are capable of evaluating models more thoroughly, it is important to extend our understanding of the techniques we are using in order to find best practices for the field.

Specifically, we found that PFA substantially outperforms KT on a variety of metrics. This result is not in agreement with obtained by Pavlik et al. (Pavlik, Cen & Koedinger, 2009), where they found that PFA is somewhat superior to the KT model fitted by brute force. We are uncertain for the reason for this divergence in results yet, it is likely due to the datasets. We also showed that the model parameters estimated by PFA were more plausible than those estimated by KT.

Within KT, we explored several aspects of the design space. We examined a brute force search as an approach for estimating model parameters, and reported results for a complete search of the space, as well as for a restricted portion of the space as has been done in previous work (Gong, Beck & Heffernan, 2010). We found that restricting the parameter space BF considers results in a moderate decrease in predictive accuracy, but completes much more quickly and results in more plausible parameters. Thus, restricting BF to only consider parameter values considered to be likely is a tradeoff. When compared to EM, we showed that BF is roughly equivalent to somewhat worse. BF-Full was about as accurate as EM, but at a cost of lower parameter plausibility and a much longer time to estimate the parameters. BF-Restricted was less accurate than EM, but had comparable run time and more plausible parameter estimates.

KT also has an issue of how to model problems that contain multiple skills. For predicting student performance, we found that a simple heuristic of using the skill with the minimum student proficiency outperformed both multiplying the skills together (i.e. an independence assumption), and making simultaneous predictions with each of the skills.

Similarly, we explored the space of design decisions to come up with some "best practices" for using PFA. We first explored the possible deficiency with PFA with negative learning rates. We found, at least for our data, this problem is relatively uncommon, and correcting it by setting negative learning rates to 0 or by using a procedure which forces the value to be non-negative, results in a negligible difference. In introducing PFA (Pavlik, Cen & Koedinger, 2009), the approach used for problems with multiple skills was to use a similar approach as knowledge tracing by splitting the training data apart into multiple copies of the item (one copy for each skill). This approach makes sense in the context of facilitating a proper comparison with KT since the train and test sets should be equal across techniques. However, we found that this methodology costs PFA some accuracy as it is naturally able to model problems with multiple skills. Researchers considering using PFA should not simply mimic the data preparation required for KT, and should take advantage of this aspect of PFA.

We also extended PFA by considering new aspects. PFA ignores ordering effects, a clue about student knowledge. We found that aging old observations resulted in a marginal improvement in performance. We also explored a version of PFA that ignored the transfer model for the problem and instead relied on all 104 skills to predict student performance on each question. Surprisingly, this

version of PFA had the highest accuracy. Researchers using PFA should consider experimenting with both data aging and bypassing their tutor's transfer model and predicting directly using all of the skills to see if those techniques improve accuracy on their data sets.

This work leaves several interesting questions unaddressed. The two major ones are possibly related:

Why are there divergent results for PFA vs. KT when compared to an analysis of Cognitive Tutor data?

Does ignoring the transfer model really result in more accurate predictions?

For divergent results, one possibility is the ASSISTments system has a poorer transfer model than the Cognitive Tutors. Another possibility is that the difference is due to some difference between ASSISTments and the Cognitive Tutors with respect to the domain, types of problems presented, or the students using the tutor. A fruitful area of future work is to rerun these experiments across multiple datasets—ideally from different research groups as tutors created by a particular group are likely to share similar properties and have inherent biases (Ioannidis, 2005).

That the version of PFA that ignored the transfer model performed the best could also be explained by having a weak transfer model. Another possibility is that there is considerable useful information contained in performance on "irrelevant" skills; in fact it was this intuition that prompted the first author to conduct the experiment. Testing this model on other data sets would help to resolve the issue. Furthermore, the approach of just using the transfer model or just using all of the skills equally can be thought of as endpoints in a continuum. Perhaps a hybrid model which separated out the impact of the skills (believed to be) exercised by this problem and the impact of all of the other skills would be even more useful?

A corollary to the above is better understanding the sensitivity of KT and PFA to various transfer models. Presumably, PFA will be more robust since it directly estimates an item parameter, providing a solid base line performance. It would be interesting to see how the techniques combine under a fair comparison with an automatically derived transfer model.

One concern the authors have is that the field will move to adopt PFA, as there is a clear reason to use KT in some circumstances: it provides a model of learning. PFA includes a parameter called learning, but it only reflects getting an item correct (and the technique is quite properly named Performance Factors Analysis). Getting an item wrong is represented by a separate parameter, which is usually negative. Clearly this value does not represent "anti-learning." The cleaner semantics of KT are an advantage for certain educational data mining activities, such as investigating which items result in the most learning. Finding a method of improving the semantics of models such as PFA, so we could have the advantages of model accuracy and model interpretability, would be a major step forward.

The marginal results for data aging point to another possible line of inquiry: considering time in the model. At present, our aging scheme simply ages data at each practice opportunity of a skill. Such an approach captures ordering effects, but neglects that a student's knowledge changes when not using the tutor. If three months have elapsed between practicing a certain skill, the student might have forgotten, or learned, something about that skill in the meantime. Thus, estimating the effect of time directly in the model would nicely augment the aging approach.

In conclusion, this paper compared PFA and KT as approaches for student modeling, and found that, for our data, PFA is a stronger approach for accuracy both of predictions and for estimated parameters. Since accuracy at predicting student performance is a good test of the accuracy of a student modeling system, system developers should consider using PFA. Educational data mining researchers should consider what it is they wish to do with the model before deciding. The use of the model, rather than the task being modeled, should drive the decision making. Also note that this issue

of clear semantics is related to, but quite distinct from parameter plausibility—an area where PFA did quite well.

Overall, we found that many decisions were not crucial. For KT users, EM vs. BF is a tossup; most researchers will probably do fine using whatever software is more easily available. Similarly, PFA's negative learning rates seem to be a minor issue and can be ignored or patched in a variety of ways. Handling of items with multiple skills was a common issue for both PFA and KT, and there are clear recommendations for both techniques there. However, for accuracy in modeling, PFA seems to be the stronger choice.

ACKNOWLEDGMENTS

This research was made possible by the US Dept. of Education, Institute of Education Science, "Effective Mathematics Education Research" program grant #R305A070440, NSF CAREER award to Neil Heffernan, the Spencer Foundation, and a Weidenmeyer Fellowship from WPI.

REFERENCES

Arroyo, I., & Woolf, B. (2005) Inferring learning and attitudes from a Bayesian Network of log file data. Proceedings of the 12th International Conference on Artificial Intelligence in Education. pp.33-40.

Baker, R.S.J.d., Corbett, A.T. & Aleven, V. (2008) More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing. Proceedings of the 9th International Conference on Intelligent Tutoring Systems, pp. 406-415

Baker, R.S.J.d. & de Carvalho, A. M. J. A.(2008) Labeling Student Behavior Faster and More Precisely with Text Replays. Proceedings of the 1st International Conference on Educational Data Mining, pp. 38-47.

Baker, R. Personal communication. (2010)

Baker, R.S., Corbett, A.T. & Koedinger, K.R. (2004) Detecting Student Misuse of Intelligent Tutoring Systems.Proceedings of the 7th International Conference on Intelligent Tutoring Systems, 531-540.

Beck, J. E. (2007) Difficulties in inferring student knowledge from observations (and why you should care). Proceedings of the AIED2007 Workshop on Educational Data Mining, Marina del Rey, CA, pp.21-30.

Beck, J. E. & Chang, K.-m. (2007) Identifiability: A Fundamental Problem of Student Modeling. Proceedings of the 11th International Conference on User Modeling.

Beck, J. E. & Mostow, J. (2008) How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students. 9th International Conference on Intelligent Tutoring Systems, Montreal, pp. 353-362.

Beck, J. E., Chang, K., Mostow, J., & Corbett, A. T. (2008) Does Help Help? Introducing the Bayesian Evaluation and Assessment Methodology. Intelligent Tutoring Systems 2008: 383-394.

Cen, H., Koedinger, K. & Junker, B. (2006) Learning Factors Analysis - A General Method for Cognitive Model Evaluation and Improvement. Proceedings of the 8th International Conference on Intelligent Tutoring Systems, Jhongli, pp. 164-175.

Chang, K.-m., Beck, J., Mostow, J.& Corbett, A. (2006) A Bayes Net Toolkit for Student Modeling in Intelligent Tutoring Systems: Intelligent Tutoring Systems: Intelligent Tutoring Systems Volume 4053/2006.

Cocea, M., Hershkovitz, A. & Baker, R.S.J.d. (2009)The Impact of Off-task and Gaming Behaviors on Learning: Immediate or Aggregate? Proceedings of the 14th International Conference on Artificial Intelligence in Education. pp.507-514.

Corbett, A.T. (2001). Cognitive computer tutors: Solving the two-sigma problem. International Conference on User Modeling. pp. 137-147.

Corbett, A. & Anderson, J. (1995) Knowledge tracing: Modeling the acquisition of procedural knowledge. User modeling and user-adapted interaction. 4: pp. 253-278.

Feng, M., Heffernan, N.T., & Koedinger, K.R. (2009). Addressing the assessment challenge in an Intelligent Tutoring System that tutors as it assesses. The Journal of User Modeling and User-Adapted Interaction. Vol 19: p243-266. 13.

Gong, Y, Beck, J. E. & Heffernan, N. T. (2010) Comparing Knowledge Tracing and Performance Factor Analysis by Using Multiple Model Fitting Procedures. Proceedings of the 10th International Conference on Intelligent Tutoring Systems. Pittsburgh.

Gong, Y., Beck, J., Heffernan, N. T.& Forbes-Summers, E. (2010)The impact of gaming (?) on learning at the fine-grained level. In Aleven, V., Kay, J & Mostow, J. (Eds) Proceedings of the 10th International Conference on Intelligent Tutoring Systems (ITS2010) Part 1. Springer. Pages 194-203.

Gong, Y., Beck, J. E. & Heffernan, N. T. (2010) Using Multiple Dirichlet distributions to improve parameter plausibility Educational Data Mining 2010. In Baker, R.S.J.d., Merceron, A., Pavlik, P.I. Jr. (Eds.) Proceedings of the 3rd International Conference on Educational Data Mining. Pages 61-70.

Gong, Y., Rai, D. Beck, J. E. & Heffernan, N. T. (2009) Does Self-Discipline impact students' knowledge and learning? In Barnes, Desmarais, Romero & Ventura (Eds) Proc. of the 2ndInternational Conference on Educational Data Mining. Pp. 61-70. ISBN: 978-84-613-2308-1.

Heathcote, A., Brown, S. & Mewhort, D. J. K. (2002) The Power Law repealed: The case for an Exponential Law of Practice. Psychonomic Bulletin & Review.

Ioannidis, J.P.A. (2005) Why Most Published Research Findings Are False. PLoS Med 2(8): e124. doi:10.1371/journal.pmed.0020124v.

Koedinger, K. R., Anderson, J. R., Hadley, W. H. & Mark, M. (1997) A.: Intelligent Tutoring Goes to School in the Big City. Int. J. Artificial Intelligence in Education.

Pardos, Z. A., Beck, J., Ruiz, C. & Heffernan, N. T. (2008) The Composition Effect: Conjunctive or Compensatory? An Analysis of Multi-Skill Math Questions in ITS. In Baker & Beck (Eds.) Proceedings of the First International Conference on Educational Data Mining. Montreal, Canada. pp. 147-156.

Pardos, Z. A. & Heffernan, N. T. (2010) Navigating the parameter space of Bayesian Knowledge Tracing models: Visualizations of the convergence of the Expectation Maximization algorithm . Proceedings of the 3rd International Conference on Educational Data Mining.

Pardos, Z. A.& Heffernan, N. T. (2010) Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing. In Paul De Bra, Alfred Kobsa, David Chin, (Eds.)The 18th Proceedings of the International Conference on User Modeling, Adaptation and Personalization. Springer-Verlag

Pavlik, P. I., Cen, H. & Koedinger, K. (2009) Performance Factors Analysis - A New Alternative to Knowledge. Proceedings of the 14th International Conference on Artificial Intelligence in Education, Brighton, UK, pp. 531-538.

Rai, D, Gong, Y & Beck, J. E. (2009) Using Dirichlet priors to improve model parameter plausibility. Proceedings of the 2nd International Conference on Educational Data Mining, Cordoba, Spain, pp141-148.

Razzaq, L., Feng, M., Nuzzo-Jones, G., Heffernan, N.T., Koedinger, K. R., Junker, B., Ritter, S., Knight, A., Aniszczyk, C., Choksey, S., Livak, T., Mercado, E., Turner, T.E., Upalekar. R, Walonoski, J.A., Macasek. M.A. & Rasmussen, K.P. (2005). The Assistment project: Blending assessment and assisting. In C.K. Looi, G. McCalla, B. Bredeweg, & J. Breuker (Eds.) Proceedings of the 12th Artificial Intelligence in Education, Amsterdam: ISO Press. pp. 555-562.

Moon, T. K. (1996) The expectation–maximization algorithm. IEEE Signal Process. Mag., vol. 13, no. 6, pp. 47–60.

van der Linden, W. J., & Hambleton, R. K. (eds.) (1997). Handbook of modern item response theory. New York, NY: Springer Verlag.