Integrating Knowledge Tracing and Item Response Theory: A Tale of Two Frameworks

*Mohammad M. Khajah[1], *Yun Huang[2], *José P. González-Brenes[3], Michael C. Mozer[1], Peter Brusilovsky[2] ⋆

[1] Department of Computer Science and Institute of Cognitive Science, University of Colorado, Boulder, CO 80309-0430, USA
{mohammad.khajah,mozer}@colorado.edu
[2] Intelligent Systems Program, University of Pittsburgh, Pittsburgh, PA 15260, USA
{yuh43,peterb}@pitt.edu
[3] Digital Data, Analytics & Adaptive Learning, Pearson Research & Innovation Network, Philadelphia, PA 19103, USA
jose.gonzalez-brenes@pearson.com

Abstract. Traditionally, the assessment and learning science communities rely on different paradigms to model student performance. The assessment community uses Item Response Theory which allows modeling different student abilities and problem difficulties, while the learning science community uses Knowledge Tracing, which captures skill acquisition. These two paradigms are complementary – IRT cannot be used to model student learning, while Knowledge Tracing assumes all students and problems are the same. Recently, two highly related models based on a principled synthesis of IRT and Knowledge Tracing were introduced. However, these two models were evaluated on different data sets, using different evaluation metrics and with different ways of splitting the data into training and testing sets. In this paper we reconcile the models' results by presenting a unified view of the two models, and by evaluating the models under a common evaluation metric. We find that both models are equivalent and only differ in their training procedure. Our results show that the combined IRT and Knowledge Tracing models offer the best of assessment and learning sciences – high prediction accuracy like the IRT model, and the ability to model student learning like Knowledge Tracing.

Keywords: Knowledge Tracing, IRT, Parameter learning

1 Introduction

In many instructional settings, students are graded by their performance on instruments such as exams or homework assignments. Usually, these instruments are made of *items* – questions, problems, parts of questions – which are graded individually. Recent interest in online education, such as Massively Open Online Courses, promises large amounts of data from students solving items over

⋆ The first three authors contributed equally to the paper.

time. The assessment and learning science communities offer two paradigms to model such data. Traditionally, the assessment community relies on *Item Response Theory* (IRT) [12] which infers individual differences amongst students and items, but it does not account for student learning over time. The learning science community uses Knowledge Tracing [2] to estimate skill acquisition as a function of practice opportunities. Although Knowledge Tracing captures student learning, it assumes that students and items do not vary in abilities or difficulties – any two items involving the same skill are assumed to be equivalent.

Empirically we know that neither models' assumptions are correct – these two paradigms are complementary. Earlier attempts towards unifying these two paradigms within the Knowledge Tracing framework only individualize students [9, 10, 16] or items[4, 11, 13] but not both. It is only recently that serious efforts have been made to integrate both student and item effects into Knowledge Tracing. Specifically, two highly related models based on a principled synthesis of Knowledge Tracing and IRT [3, 6] were proposed. The two models were evaluated on different data sets, using different evaluation metrics and with different ways of splitting the data into training and testing sets. In this paper we reconcile the models' results by presenting a unified view of the two models, and by evaluating the models under a common evaluation metric.

The rest of this paper is organized as follows. Section 2 describes the two recent methods that unify Knowledge Tracing and Item Response Theory. Section 3 provides empirical evaluation. Section 4 concludes.

2 Methods

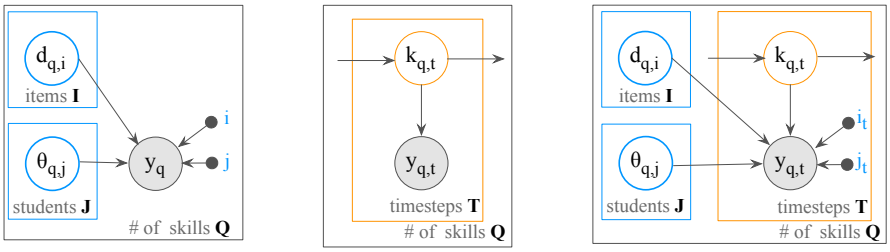
Recently, two models were proposed independently which synthesize Knowledge Tracing and IRT: FAST [3] and LFKT [6]. Although the two models are described in somewhat different terms, they are nearly equivalent, with the key difference being their training method. We now present a unified view of the two models, and then explain their parameter estimate procedures.

2.1 Model Structure

Figure 1 uses plate notation to describe IRT, Knowledge Tracing, and the combined model. In plate notation, the clear nodes represent latent variables; the light gray nodes represent variables that are observed only in training; dark nodes represent variables that are both visible in training and testing; plates represent repetition of variables. We omit drawing the parameters and priors.

Figure 1a shows the plate diagram of the Rasch model, the simplest IRT model. Rasch treats each skill q independently, and can be modeled using logistic regression with binary variables indicators for each student i and each item j. The regression coefficients θ_q and d_q of the binary features can then be interpreted as student ability and item difficulty, respectively. The binary observation variable (y_q) represents whether the student gets an item correct:

$$p(y_q) = \text{logistic}(\theta_{q,i}, d_{q,j}) = \frac{1}{1 + e^{-(\theta_{q,j} + d_{q,i})}} \tag{1}$$



(a) Item Response Theory (b) Knowledge Tracing (c) Combined IRT and KT

Fig. 1: Plate diagram notation for different student models

Figure 1b describes the Knowledge Tracing model. Knowledge Tracing uses Hidden Markov Models (HMMs)to infer the binary latent student knowledge state ($k_{q,t}$) indicating whether the skill has been mastered at the t^{th} learning opportunity of skill q. The *transition probabilities* between latent states are often referred as learning and forgetting probabilities, and the *emission probabilities* are commonly referred as guess and slip probabilities. The binary variable $y_{q,t}$ represents whether the student gets an item correct:

$$P(y_{q,t}|y_{q,1}...y_{q,t-1}) = \sum_{\substack{l \in \{\text{mastered,} \\ \text{not mastered}\}}} \overbrace{P(y_{q,t}|k_t = l)}^{emission\ probability} \cdot P(k_t = l|y_{q,1}...y_{q,t-1}) \quad (2)$$

Figure 1c shows the combined model. It replaces the emissions with IRT:

$$P(y_{q,t}|y_{q,1}...y_{q,t-1}) = \sum_{\substack{l \in \{\text{mastered,} \\ \text{not mastered}\}}} \overbrace{\text{logistic}(d_{q,i_t}, \theta_{q,j_t}, c_{q,l})}^{\text{IRT}} \cdot P(k_t = l|y_{q,1}...y_{q,t-1}) \quad (3)$$

Here, the logit is parametrized by the difficulty d of the item i, the ability θ of student j and a bias c that is specific to whether the student has mastered the skill. Both Knowledge Tracing and IRT can be recovered from the combined model with different choices of parameter values. For example, when the abilities and difficulties are zero, the combined model is equivalent to Knowledge Tracing. When the bias terms are the same (i.e., $c_{\text{not mastered}} = c_{\text{mastered}}$), we get IRT.

2.2 Parameter Learning

We now briefly review two recent proposals to learn the combined model. A thorough discussion can be found elsewhere [3, 6]. González-Brenes et al. [3] use a recent variant of the EM algorithm [1] that allows learning HMMs with arbitrary features. Although the original framework allows general features to be incorporated into Knowledge Tracing, the model becomes equivalent to our combined model when limiting the features to IRT features only.

Table 1: Basic Dataset Statistics

	Geometry	**Physics**	**Statics**	**QuizJET**
Trials	5,104	110,041	189,297	6,549
Students	59	66	333	110
Problems	139	4,816	1,223	95
Skills	18	652	156	19
Mean Seq. Length	8.0	4.5	6.0	4.7
Mean Correct	75%	83%	77%	60%

Alternatively, Khajah et al. [6] use Bayesian estimation techniques to learn the combined model. They used slice sampling, a MCMC algorithm that generates samples of the joint posterior distribution of the model. This allows using priors on abilities and difficulties which can be used to generalize to unseen students and items. Also, their model allows to fit student ability parameters across different skills – using data from a student's performance on one skill to predict their performance on a different skill.

3 Results

We evaluate our student models by how accurately they predict future student performance. We operationalize predicting future student performance as the classification task of predicting which students solved correctly the items in a held-out set. We evaluate them using a popular machine learning metric, the *Area Under the Curve* (AUC) of the Receiver Operating Characteristic (ROC) curve. The AUC is an overall summary of diagnostic accuracy. AUC equals 0.5 when the ROC curve corresponds to random chance and 1.0 for perfect accuracy. We report the 95% confidence intervals with an implementation of the bootstrap hypothesis test method (`http://subcortex.net/research/code/`), a method that corrects for the non-independence of the points of the ROC.

We use data from four different intelligent tutoring systems: the Geometry Cognitive Tutor [7], the Andes Physics Tutor [15], OLI Statics [14] and QuizJET Java programming tutor [5]. The first three datasets are available on the PSLC Datashop [8]. Table 1 shows a summary of their descriptive statistics.

We divide each dataset into skill-specific subsets consisting of the sequence of trials for each student involving items that require a particular skill. We refer to these sequences as student-skill sequences. If multiple skills are associated with a item, we treat the combination of skills as one unique skill. The last 20% of trials from each sequence were placed in the testing set. Thus, generalization to new skills is not required. For each trial, we compute the prediction (probability of a correct response). Predicted outcomes in the test set are then used to calculate the AUC.

Bar heights in figure 2 are the AUC values for each model and error bars represent 95% confidence intervals. We evaluated the Bayesian and Maximum Likelihood versions of knowledge tracing, IRT, and the combined model. On the smallest dataset, geometry all models perform similarly during testing. On the next two larger datasets, QuizJET and physics, IRT and the combined models perform similarly, beating knowledge tracing. Neither IRT or the combined models emerge as a clear winner. However, on the largest dataset, statics, the Bayesian combined model outperforms all other models significantly. In this dataset, IRT trained using Maximum Likelihood beats the Bayesian version, which might be due to the effects of strong priors on student abilities and item difficulties in the MCMC-trained version. This would also explain why the Bayesian IRT would gain advantage in smaller datasets where the priors influence the most.

In three datasets, the Bayesian version of Knowledge Tracing beats Maximum Likelihood. Our hypothesis is that MCMC training used in Bayesian estimation is more effective at avoiding local optima. We do not think it is due to the use of priors, because we used uninformative priors.

We hypothesize that the reason why the combined model does not outperform IRT is because of the order in which items are presented to students. Specifically, if the items are presented in a relatively deterministic order, the item's position in the sequence of trials is confounded with the item's identity. IRT can exploit such a confound to implicitly infer performance levels as a function of experience, and therefore would have the same capabilities as the combined model which performs explicit inference of student knowledge state. To investigate this, we compute the mean order in which items are presented to students. In Figure 3, the horizontal axis ranks item indices by their mean presentation order, and the vertical axis is the mean order in which items are shown to students. Flat horizontal lines in this plot suggest random item ordering but they may also confound the case where students are assigned to only one item from a set of items. On geometry, we don't see any flat sections which suggests fixed item ordering. Next, the QuizJET dataset exhibits periodic flat sections, but these could be due to students being assigned to single specific items out of sets of items. On the physics and statics datasets, we see a flat line followed by a curve which suggests an initial random assignment of items followed by more structured item ordering. So, there is less information overlap between student learning and item difficulties in the physics and statics datasets, thereby allowing the combined model to put its extra parameters to good use. We plan to carry out more rigours tests of this conclusion in the future.

One of the goals of intelligent tutoring systems is to personalize learning. For example, teaching a new skill instead of practicing one that the student has mastered. Personalization requires student models that capture skill acquisition over time, which is possible with Knowledge Tracing and the combined model, but not with IRT. We calculate the estimated learning of a student as the probability of mastery at the last practice opportunity minus the probability of mastery at the first practice opportunity for each student-skill sequence ($p(k_T = \text{master}) - p(k_0 = \text{master})$). A value of 0 indicates no difference whilst

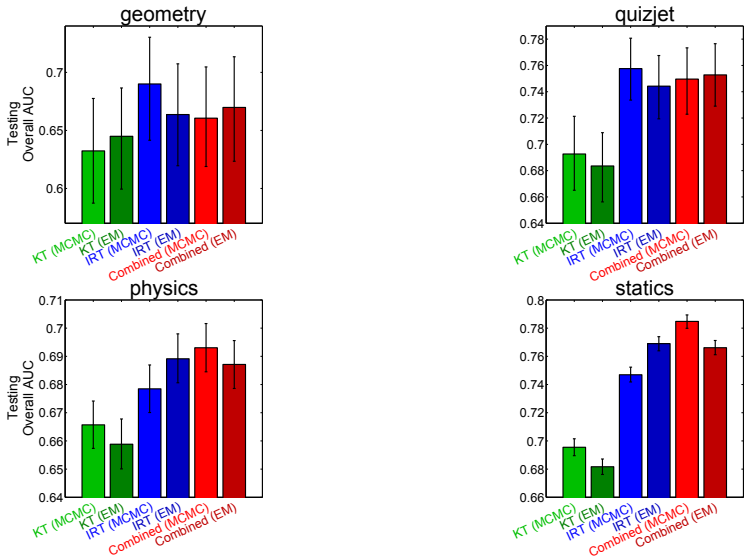


Fig. 2: Test set AUC scores of six models on four datasets. Higher values indicate better performance. Error bars correspond to 95% confidence intervals.

a value of 1 indicates maximum difference. Figure 4 shows the mean difference in the estimated student learning over all student-skill sequences. For clarity, we omit to draw the estimate of IRT, which assumes no learning occurs. This result suggests that the improved performance of the combined model could be useful in intelligent tutoring systems.

4 Conclusion

In this paper we investigate two recent alternatives that integrate Knowledge Tracing and IRT. We discover that both models are in fact equivalent and differ only in their training procedure – using either Maximum Likelihood or Bayesian Estimation. We compare both training procedures, Maximum Likelihood and Bayesian estimation, using the same four datasets, cross validation splits and evaluation metric. We find out that both training methods have similar performance, with a small advantage to the Bayesian method in the largest dataset we used. Future work may investigate why this is the case. The combined model only outperforms IRT in one dataset. In future work we will investigate whether the lack of improvement is due to a confound of item identity and position in the sequence of trials when nearly deterministic trial sequences are presented

We find that the combined method persistently outperforms Knowledge Tracing, and unlike IRT, it is able to model student learning. Future work may evaluate how useful the combined model is for personalizing learning in an intelligent tutoring system.

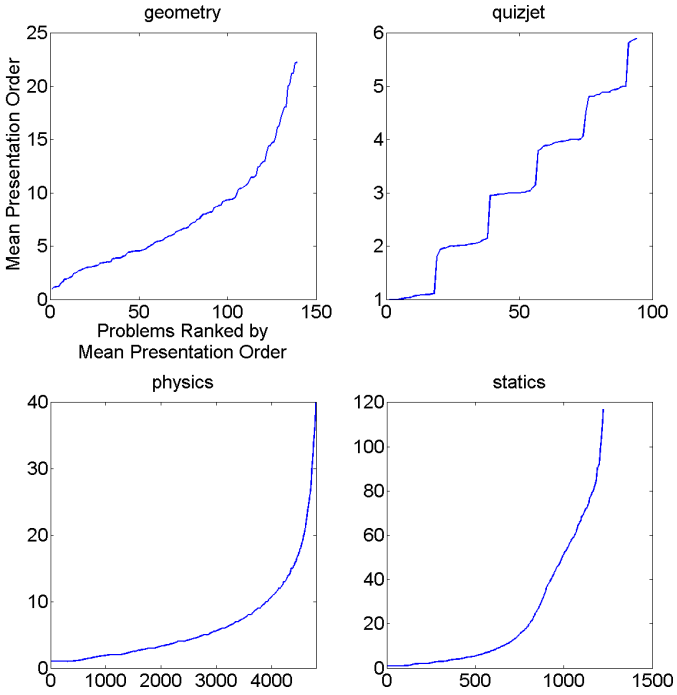


Fig. 3: Mean presentation order for each item in all four datasets (plates). The horizontal axis ranks item indices by their mean presentation order. The vertical axis is the mean order in which items are shown to students.

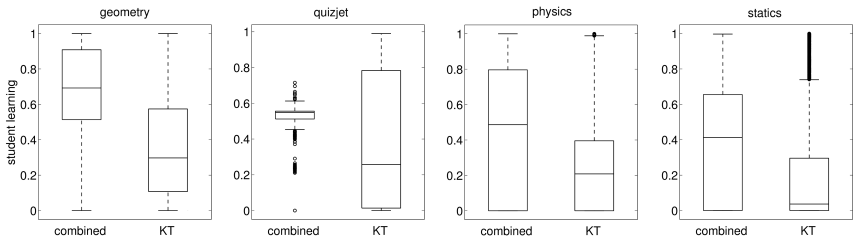


Fig. 4: Boxplot of the estimated student learning of Knowledge Tracing and the combined model. We omit IRT, because it always assumes no learning.

Bibliography

[1] Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590. Association for Computational Linguistics, 2010.

[2] AlbertT. Corbett and JohnR. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1995. ISSN 0924-1868. doi: 10.1007/BF01099821. URL `http://dx.doi.org/10.1007/BF01099821`.

[3] 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 *Submitted to the 7th International Conference on Educational Data Mining (2014)*.

[4] Sujith M. Gowda, Jonathan P. Rowe, Ryan Shaun Joazeiro de Baker, Min Chi, and Kenneth R. Koedinger. Improving models of slipping, guessing, and moment-by-moment learning with estimates of skill difficulty. In Mykola Pechenizkiy, Toon Calders, Cristina Conati, Sebastián Ventura, Cristóbal Romero, and John C. Stamper, editors, *EDM*, pages 199–208. www.educationaldatamining.org, 2011. ISBN 978-90-386-2537-9.

[5] I-Han Hsiao, Sergey Sosnovsky, and Peter Brusilovsky. Adaptive navigation support for parameterized questions in object-oriented programming. In *Learning in the Synergy of Multiple Disciplines*, pages 88–98. Springer, 2009.

[6] M. Khajah, R. M. Wing, R. V. Lindsey, and M. C. Mozer. Integrating latent-factor and knowledge-tracing models to predict individual differences in learning. In *In submission*, 2014.

[7] Ken Koedinger. Geometry area (1996-97), February 2014. URL `https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=76`.

[8] 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.

[9] Jung In Lee and Emma Brunskill. The impact on individualizing student models on necessary practice opportunities. In Kalina Yacef, Osmar R. Zaïane, Arnon Hershkovitz, Michael Yudelson, and John C. Stamper, editors, *Proceedings of the 5th International Conference on Educational Data Mining*, pages 118–125, Chania, Greece, 2012. www.educationaldatamining.org. URL `http://educationaldatamining.org/EDM2012/uploads/procs/Full_Papers/edm2012_full_11.pdf`.

[10] Zachary A. Pardos and Neil T. Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *Proceedings of the 18th international conference on User Modeling, Adaptation, and Personalization*, UMAP'10, pages 255–266, Berlin, Heidelberg,

2010. Springer-Verlag. ISBN 3-642-13469-6, 978-3-642-13469-2. URL http://dx.doi.org/10.1007/978-3-642-13470-8_24.

[11] ZacharyA. Pardos and NeilT. Heffernan. Kt-idem: Introducing item difficulty to the knowledge tracing model. In JosephA. Konstan, Ricardo Conejo, Jos L. Marzo, and Nuria Oliver, editors, *User Modeling, Adaption and Personalization*, volume 6787 of *Lecture Notes in Computer Science*, pages 243–254. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-22361-7. doi: 10.1007/978-3-642-22362-4_21. URL http://dx.doi.org/10.1007/978-3-642-22362-4_21.

[12] G. Rasch. Probabilistic models for some intelligence and attainment tests. In *Paedagogike Institut, Copenhagen.*, 1960.

[13] Sarah Schultz and Trenton Tabor. Revisiting and extending the item difficulty effect model. In *In Proceedings of the 1st Workshop on Massive Open Online Courses at the 16th Annual Conference on Artificial Intelligence in Education*, pages 33–40, Memphis, TN., 2013.

[14] Paul Steif and Norman Bier. Oli engineering statics - fall 2011, February 2014. URL https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=507.

[15] Kurt VanLehn. Usna physics fall 2006, February 2014. URL https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=126.

[16] Michael Yudelson, Kenneth R. Koedinger, and Geoffrey J. Gordon. Individualized bayesian knowledge tracing models. In *Artificial Intelligence in Education - 16th International Conference (AIED 2013)*, pages 171–180, Memphis, TN, 2013. Springer.