What and When do Students Learn? Fully Data-Driven Joint Estimation of Cognitive and Student Models

José P. González-Brenes† and Jack Mostow*
Project LISTEN
www.projectlisten.com
†Language Technologies Institute, *Robotics Institute
Carnegie Mellon University, Pittsburgh, PA
{joseg, mostow}@cmu.edu

ABSTRACT

We present the Topical Hidden Markov Model method, which infers jointly a cognitive and student model from longitudinal observations of student performance. Its cognitive diagnostic component specifies which items use which skills. Its knowledge tracing component specifies how to infer students' knowledge of these skills from their observed performance. Unlike prior work, it uses no expert engineered domain knowledge — yet predicts future student performance in an algebra tutor as accurately as a published expert model.

Keywords

knowledge component discovery, student modeling, cognitive diagnostic model, knowledge tracing

1. INTRODUCTION

Assessing students' skills from their performance requires a *cognitive diagnostic model* specifying which observed items require which skills (sometimes called knowledge components), and a *student model* that infers how well students know each skill, based on their performance on items requiring that skill. For example, a cognitive diagnostic model for a reading tutor that listens to children read aloud might model the graphophonemic patterns in a word as distinct skills. Cognitive diagnostic models are typically engineered by human domain experts at considerable expense. Methods to infer them automatically from student performance data have been restricted to static instruments such as exams or homework assignments administered only once or twice. However, intelligent tutorial decisions require a student model that traces changes in student skills dynamically over time. This paper presents and evaluates the novel data-driven Topical HMM method to discover a cognitive diagnostic model and a student model simultaneously.

.

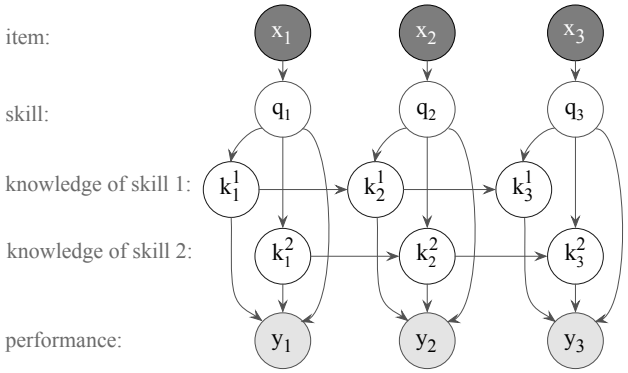

Figure 1: Unrolled example of Topical HMM with two skills ($\mathbf{S} = 2$), for a single user ($\mathbf{U} = 1$) with three time steps ($\mathbf{T_1} = 3$). Student indices, parameters (Q, K, D) and priors (α, τ, ω) are omitted for clarity. Dark gray variables are observable during both training and testing. Light gray variables are visible only during training. White variables are never observed (latent).

2. TOPICAL HIDDEN MARKOV MODEL

Topical HMM treats the skills required by a sequence of observed items as latent topics. We use a mixed membership model to represent the latent skill(s) required by an item. That is, we represent the item as requiring a single skill whose identity is uncertain but has a specified probability distribution, which we interpret as specifying the relative weight of each skill for the item. Figure 1 unrolls this graphical model for two skills. The absence of connections between knowledge nodes for different skills assumes no transfer between skills, i.e., the student's knowledge of a skill can change only when the student encounters an item that requires the skill. This assumption makes possible an efficient Gibbs sampler not described here.

Algorithm 1 specifies Topical HMM's generative story. It has hyper-parameters, variables, parameters, and priors.

Topical HMM's hyper-parameters are given or tuned:

- $\mathbf{S}$ is the number of skills in the model.
- $\mathbf{U}$ is the number of users (students).
- $\mathbf{T_u}$ is the number of time steps student u practiced.
- $\mathbf{M}$ is the number of items. For example, in the case of a reading tutor, $\mathbf{M}$ may represent the vocabulary size.

In a tutor that creates items dynamically, **M** is the number of templates from which items are generated.

- **L** is the number of levels of knowledge of a skill, typically 2 (knowing it or not). To distinguish novice, medium, and expert proficiency, we would use $\mathbf{L} = 3$.

Topical HMM's variables correspond to nodes in Figure 1:

- $x_{u,t}$ is the item the student u encountered at time t.
- $q_{u,t}$ is a latent random variable specifying the skill(s) required for item $x_{u,t}$.
- $k^s_{u,t}$ is a variable that takes values from $1 \dots L$ to represent the level of knowledge of skill s. There is a Markovian dependency across time steps: if skill s is known at time $t-1$, it is likely to still be known at time t.
- $y_{u,t}$ represents student performance as a binary variable (correct or not), observed only during training.

Topical HMM's parameters specify the distributions of these variables. Since we take a fully Bayesian approach, we model parameters as random variables:

- $Q^{x_{u,t}}$ is the cognitive diagnostic model. It represents the skill(s) required for item $x_{u,t}$ as a multinomial $Q^{x_{u,t}}$ to model soft membership. For example, $Q^{x_{u,t}} = [0.75, 0.25, 0, 0]$ means that item $x_{u,t}$ depends mostly on skill 1, less on skill 2, and not at all on skills 3 or 4. Unlike prior work where the mapping of items to skills must be given, Topical HMM allows Q to be hidden, i.e. discovered entirely from data.
- $K^{s,l}$ is a multinomial that specifies the transition probabilities from knowledge state l of skill s to other knowledge states.
- $D^{s,l}$ is a binomial that specifies the emission (output) probability of a correct answer given the student's proficiency level l on the required skill s.

Topical HMM uses Dirichlet priors α, τ, ω for its parameters.

Algorithm 1 Generative story of Topical HMM

Require: A sequence of item identifiers $x_{1 \dots T_u}$ for **U** users, number of skills **S**, number of student states **L**, number of items **M**

function TOPICAL HMM($x_1 \dots x_t, \mathbf{S}, \mathbf{U}, \mathbf{L}, \mathbf{M}$)
  ▷ Draw parameters from priors:
  for each skill $s \leftarrow 1$ to **S** **do**
    for each knowledge state $l \leftarrow 1$ to **L** **do**
      Draw parameter $K^{s,l} \sim$ Dirichlet($\tau^{s,l}$)
      Draw parameter $D^{s,l} \sim$ Dirichlet($\omega^{s,l}$)
  for each item $m \leftarrow 1$ to **M** **do**
    Draw $Q^m \sim$ Dirichlet(α)
  ▷ Draw variables from parameters:
  for each student $u \leftarrow 1$ to **U** **do**
    for each timestep $t \leftarrow 1$ to $\mathbf{T}_u$ **do**
      Draw skill $q_{u,t} \sim$ Multinomial($Q^{x_{u,t}}$)
      for $s \leftarrow 0$ to **S** **do**
        if $s = q_{u,t}$ **then**
          ▷ knowledge state could change:
          $k'' \leftarrow k^s_{u,t-1}$ ▷ previous time step
          Draw $k^s_{u,t} \sim$ Multinomial($K^{s,k''}$)
        else
          ▷ knowledge state can't change:
          $k^s_{u,t} \leftarrow k^s_{u,t-1}$
      $q' \leftarrow q_{u,t}$ ▷ current skill
      $k' \leftarrow k^{q_{u,t}}_{u,t}$ ▷ current knowledge state
      Draw performance $y_{u,t} \sim$ Multinomial($D^{q',k'}$)

3. EVALUATION

We use data collected by the Bridge to Algebra Cognitive Tutor® [8] from 123 students, each of whom encountered an average of 340.7 items (minimum 48, maximum 562, median 341), for a total of 41,911. The data is unbalanced: over 80% of the items were correct.

We randomly partition the data into three sets with non-overlapping students – a training set with 97 students, and development and test sets with 13 students each. We use the development set to tune hyper-parameters and select the number of skills to model the data. We use the training set exclusively for learning the parameters of the model, and we only report results on the development or test set. To avoid tuning on test data, we used the test set only once, just before writing this paper.

The data set contains data from 893 different problems. Each problem consists of a sequence of one or more steps, and it is at this level that we do our analysis. We consider the different steps to be the items the student encounters. Students did not follow the curriculum in the same order; the tutor decided which problems to assign in what order, and the students chose the order to do the steps in each problem. To name items consistently across students, we named each item by concatenating the tutor-logged problem name and step name, yielding 5,233 distinct items.

We evaluate cognitive diagnostic model by how accurately they predict future student performance. We operationalize predicting future student performance as the classification task of predicting whether students correctly solved the items on a held-out set. This paper focuses on predicting performance on unseen students. To make predictions on the development and test set, we use the history preceding the time step we want to predict. To speed up computations, we predict up to the up to the 200^{th} time step in the test set. Since we run evaluations multiple times in the development set, we only predict up to the 150^{th} time step. Therefore, our development and test sets have 1950 and 2600 observations respectively.

We evaluate the classifiers' predictions using a popular data mining metric, the Area Under the Curve (AUC) of the Receiver Operating Characteristic (ROC) curve. The ROC evaluates a classifier's performance across the entire range of class distribution and error costs. An AUC of 1 represents a perfect classifier; an AUC of 0.5 represents a useless classifier, regardless of class imbalance. AUC estimates can be interpreted as the probability that the classifier will assign a higher score to a randomly chosen positive example than to a randomly chosen negative example.

The manual expert cognitive diagnostic model was developed and refined by two cognitive scientists and a teacher over four years. They first identified 76 different categories of items, and then determined that students would need fifty different skills to answer them. The manual model includes some items that use multiple skills.

When we use Topical HMM with a manually designed model, we initialize the parameter Q of Topical HMM with

the expert model and do not update its values. In the case that the expert decided that an item uses multiple skills, we assign uniform weight to each skill even though the experts assumed a conjunctive model. Topical HMM cannot handle a conjunctive cognitive diagnostic model.

We now describe the values we use for the priors' hyperparameters α, τ, and ω.

- **Sparse cognitive model**. We encourage sparsity on the cognitive diagnostic model parameter (Q), motivated by the assumption that items use only a few skills. We set $\alpha = 0.1$, because when the value of the hyper-parameter of a Dirichlet prior is below one, the samples are sparse multinomials. For example, $Q_i = [1, 0, 0, 0]$ is a sparse multinomial, that represents that item i depends on skill 1, but not on skill 2, 3 or 4.
- **Practice helps learning, and there is no forgetting**. Manipulating the magnitude of the hyperparameters τ and ω allows us to select the strength of the prior belief that students transition to a higher level of knowledge, and that they do not go back to the previous level. We use cross validation to select the magnitude of these hyperparameters with values 10 or 100.

For our experiments, we initialize the model randomly and then collect 2,000 samples from a Gibbs Sampling Algorithm. We discard the first 500 samples as a burn-in period. To infer future student performance, we save the last 1,500 samples, averaging over the samples and calculating the Maximum A Posteriori (MAP) estimate.

We compare the performance of these methods:

- **HMM**. Can we find evidence of multiple skills? Topical HMM should perform better than a cognitive model that assigns all of the items to a single skill. We run Knowledge Tracing [4] with a cognitive diagnostic model that has only one skill in total. This approach is equivalent to a single HMM.
- **Student Performance**. What is the effect of students' individual abilities? We predict that the likelihood of answering item at time t correctly is the percentage of items answered correctly up to time $t-1$. Intuitively, this is the student's "batting average".
- **Random cognitive diagnostic model**. Does the cognitive diagnostic model matter? We create a random cognitive diagnostic model with five skills and assign items randomly to one of five categories. We then train Topical HMM to learn the student model (transition and emission probabilities), without updating the cognitive diagnostic model.
- **Item difficulty**. What is the classification accuracy of a simple classifier? We use a classifier that predicts the likelihood of answering item x as the mean performance of students in the training data on item x. Note that this classifier does not create a cognitive diagnostic model.
- **Manual cognitive diagnostic model**. How accurate are experts at creating a cognitive diagnostic models? We use Topical HMM with the 50-skill cognitive diagnostic model designed by an expert.
- **Data-driven cognitive diagnostic model**. We initialize Topical HMM with the best model discovered using the development set (with 5 skills).

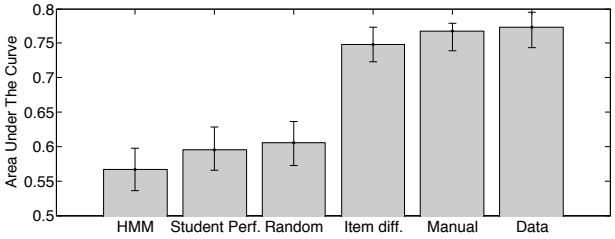

Figure 2: Test set AUC performance of different models

Figure 2 shows the AUC of the different methods applied to the test set, with 95% confidence intervals calculated with an implementation of the Logit method [1]. Our data-driven model with five skills outperforms all of the other models, with an AUC of 77.28. Because the confidence intervals do not overlap, we can conclude with 95% confidence that our data-driven model is significantly better than assuming a cognitive diagnostic model with a single skill (HMM), using the student's "batting average" (Student Perf.), or assigning items to skills randomly (Random). The confidence intervals for the data-driven cognitive diagnostic model, the manually engineered cognitive diagnostic model, and the item difficulty approach overlap, with AUC scores of 77.28, 76.71 and 74.76 respectively.

4. RELATION TO PRIOR WORK

This section relates Topical HMM to prior work in automatic discovery of cognitive diagnostic models and student models. In psychometrics, the branch of psychology and education concerning educational statistics, matrix factorization methods have been applied to discover a cognitive diagnostic model from static assessment instruments such as a single exam, or a homework assignment. A survey of previous approaches to automatic discovery of cognitive diagnostic models can be found elsewhere [13]; popular approaches include Item Response Theory [10], and matrix factorization techniques such as Principal Component Analysis, Non-Negative Matrix Factorization [5, 13], and the Q-Matrix Method [1]. These methods can help explain what skills students have mastered, but they ignore the temporal dimension of data. Unlike Topical HMM, these approaches do not discover a clustering of items to skills per se: performance is based on continuous latent traits. More specifically, matrix factorization techniques predict student performance as a combination of latent user traits, and latent item difficulty traits (skills) that may be multidimensional. Moreover, matrix factorization techniques cannot be applied to the problem of predicting performance of unseen students, because they require the latent user trait matrix. This problem also carries over for higher dimension factorization techniques, such as tensor factorization [12].

Learning Factors Analysis [3] uses temporal data, but requires initial knowledge to improve upon. Dynamic Cognitive Tracing [7] proposed a fully automatic method, but did not scale due to memory use exponential in the number of items and runtime exponential in the number of skills. Moreover, Dynamic Cognitive Tracing was only tested on synthetic data.

[1] `http://www.subcortex.net/research/code/area_under_roc_curve`

To our knowledge, we are the first ones to take time into consideration to estimate a cognitive diagnostic model from data of real students interacting with a tutor.

Knowledge Tracing [4] is a popular method to model students' changing knowledge during skill acquisition. It requires (a) a cognitive diagnostic model that maps each item to the skill(s) required, and (b) logs of students' correct and incorrect answers as evidence of their knowledge of particular skills. Knowledge Tracing can be formulated as a graphical model: items that belong to the same skill are grouped into a single sequence, and an HMM is trained for each sequence. The observable variable is the performance of the student solving the item, and the hidden state is a binary latent variable that represents whether the student knows the skill. Topical HMM generalizes Knowledge Tracing, which assumes the cognitive diagnostic model is known and each item uses exactly one skill. Topical HMM discovers the cognitive diagnostic model automatically and is more flexible since it allows more than one skill per item.

Attempts to use tensor factorization – matrices with more than two dimensions – to model student learning have been limited [12] as they require all students and items to be seen during training, which is often not feasible.

Other approaches to student modeling also exist. Performance Factors Analysis [6] predicts student performance based on item difficulty and student performance. Learning Decomposition [2] uses non-linear regression to determine how to weight the impact of different types of practice opportunities relative to each other. Parameter Driven Process for Change [11] is able to use different student modeling techniques, such as Knowledge Tracing or NIDA [9], to group students with similar response or skill patterns over time.

5. CONCLUSIONS AND FUTURE WORK

Our main contribution is a novel method, Topical HMM, which discovers cognitive and student models automatically. A difficulty of modeling real student data is sparsely observed students, items and skills. Unlike some prior methods, Topical HMM discovers cognitive diagnostic models that generalize to unseen students. Our work is also the first automatic approach to discover a cognitive diagnostic model from real student data collected over time.

Previous work on automatic discovery of cognitive diagnostic models from static data was successful in distinguishing between broad areas (i.e., French and Math), but not finer distinctions within an area [13, 5]. Given that we were able to discover different skills within an algebra tutor data set we are optimistic about this line of research. In future work we are interested in assessing the interpretability of the cognitive diagnostic models discovered by Topical HMM. A limitation of this study is that we evaluated our approach on only one dataset. Future work may test Topical HMM on more data sets from real students.

Acknowledgements

This work was supported in part by the Pittsburgh Science of Learning Center, the Costa Rican Ministry of Science and Technology (MICIT), and National Science Foundation Grant IIS1124240 to Carnegie Mellon University. The opinions expressed are those of the authors and do not necessarily represent the views of PSLC, MICIT or the National Science Foundation. We thank the educators, students, and LISTENers who helped in this study and the reviewers for their helpful comments.

6. REFERENCES

[1] T. Barnes, D. Bitzer, and M. Vouk. Experimental Analysis of the Q-Matrix Method in Knowledge Discovery. In M.-S. Hacid, N. Murray, Z. Ras, and S. Tsumoto, editors, *Foundations of Intelligent Systems*, volume 3488 of *Lecture Notes in Computer Science*, pages 11–41. Springer Berlin / Heidelberg, 2005.

[2] J. Beck and J. Mostow. How who should practice: Using learning decomposition to evaluate the efficacy of different types of practice for different types of students. In B. Woolf, E. Ameur, R. Nkambou, and S. Lajoie, editors, *Intelligent Tutoring Systems*, volume 5091 of *Lecture Notes in Computer Science*, pages 353–362. Springer Berlin / Heidelberg, 2008.

[3] H. Cen, K. Koedinger, and B. Junker. Learning factors analysis: A general method for cognitive model evaluation and improvement. In M. Ikeda, K. Ashley, and T.-W. Chan, editors, *Intelligent Tutoring Systems*, volume 4053 of *Lecture Notes in Computer Science*, pages 164–175. Springer Berlin / Heidelberg, 2006.

[4] A. Corbett and J. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.

[5] M. Desmarais. Conditions for Effectively Deriving a Q-Matrix from Data with Non-negative Matrix Factorization. In M. Pechenizkiy and T. Calders and C. Conati and S. Ventura and C. Romero and J. Stamper, editor, *Proceedings of the 4th International Conference on Educational Data Mining*, pages 169–178, 2011.

[6] Y. Gong, J. Beck, and N. Heffernan. Comparing knowledge tracing and performance factor analysis by using multiple model fitting procedures. In V. Aleven, J. Kay, and J. Mostow, editors, *Intelligent Tutoring Systems*, volume 6094 of *Lecture Notes in Computer Science*, pages 35–44. Springer Berlin / Heidelberg, 2010.

[7] J. P. González-Brenes and J. Mostow. Dynamic Cognitive Tracing: Towards Unified Discovery of Student and Cognitive Models. In Yacef, K., Zaïane, O., Hershkovitz, H., Yudelson, M., and Stamper, J. , editor, *Proceedings of the 5th International Conference on Educational Data Mining*, pages 49–56, 2012.

[8] K. Koedinger, R. Baker, K. Cunningham, A. Skogsholm, B. Leber, and J. Stamper. *A data repository for the community: The PSLC DataShop*. CRC Press, Boca Raton, FL, 2010.

[9] E. Maris. Estimating multiple classification latent class models. *Psychometrika*, 64(2):187–212, 1999.

[10] G. Rasch. On general laws and the meaning of measurement in psychology. In *Proceedings of the Fourth Berkeley symposium on mathematical statistics and probability*, volume 4, pages 321–333. University of California Press Berkeley, CA, 1961.

[11] C. Studer, B. Junker, and H. Chan. Incorporating learning into the cognitive assessment framework. In *Presentation at Annual Meeting of the Society for Research on Educational Effectiveness*, Washington, D.C., March 2012.

[12] N. Thai-Nghe, L. Drumond, A. Krohn-Grimberghe, and L. Schmidt-Thieme. Recommender system for predicting student performance. *Procedia Computer Science*, 1(2):2811 – 2819, 2010.

[13] T. Winters, C. Shelton, T. Payne, and G. Mei. Topic extraction from item-level grades. In J. Beck, editor, *American Association for Artificial Intelligence 2005 Workshop on Educational Datamining*, Pittsburgh, PA, 2005.