Incorporating Student Response Time and Tutor Instructional Interventions into Student Modeling

Chen Lin, Shitian Shen, Min Chi
Department of Computer Science
North Carolina State University
{clin12, sshen, mchi}@ncsu.edu

ABSTRACT

Bayesian Knowledge Tracing (BKT) is one of the most widely adopted student-modeling methods. It uses *performance* (*incorrect*, *correct*) to infer student knowledge state (*unlearned*, *learned*). However, performance can be noisy and thus we explored another type of observations – student response time. Furthermore, we proposed Intervention Bayesian Knowledge Tracing (Intervention-BKT) which can incorporate multiple types of instructional interventions into the conventional BKT model. Our results show that for next-step performance predictions, Intervention-BKT is more effective than BKT; whereas to predict students' post-test scores, including student response time would yield better result than using performance alone.

Keywords

Hidden Markov Model; Input Output Hidden Markov Model; Student Modeling; Response Time

1. INTRODUCTION

Bayesian Knowledge Tracing (BKT) [5] is a widely used student-modeling approach for Intelligent Tutoring Systems (ITSs). In this paper, we extended the conventional BKT model by leveraging student response time and tutor instructional interventions. The conventional BKT model infers students' hidden knowledge states mainly from their performance (i.e., *correct*, *incorrect*). Nevertheless, student performance can be noisy because many ITSs allow students to refer to external resources for information. The ability to solicit help from external resources obscures the fact of whether a student has truly learned or not. On the other hand, ever since the mid-1950s, response time has been used as a preferred dependent variable in cognitive psychology [13]. It has mainly been used to assess student learning because response time can indicate how active and accessible student knowledge is. For example, it is shown that response time reveals student proficiency [11] and there was a significant negative correlation between student average response time and student final exam score taken at the end of the semester [7]. To build effective student modeling, in this paper we explored three types of observations: the conventional *performance*, the proposed *student response time*, and the *combined* which uses both.

To further improve our model, we incorporate multiple types of instructional interventions into the conventional BKT framework. Instructional interventions indicate actions initiated by the system to guide student learning activity. We proposed a new approach called Intervention-Bayesian Knowledge Tracing (Intervention-BKT). To determine whether introducing response time and instructional intervention lead to better student models, we constructed nine model variations {BKT, BKT (without tell), Intervention-BKT} × {*performance*, *time*, *combined*}. These nine model variations were tested on two important prediction tasks: 1) to predict students' next step performance and 2) to predict their post-test scores.

2. RELATED WORK

In recent years a variety of BKT extensions have been proposed. For example, Pardos and Heffernan [10] added problem nodes to capture item difficulty. Their model achieved performance gain on the ASSISTments dataset, but not on the Cognitive Tutor dataset. In addition, Pardos and Heffernan proposed Prior Per Student model [9], which adds a multinomial node representing student's incoming competence. They showed their model performed better than the BKT. Yudelson et al [15] later revisited the same problem and showed parametrizing student speed of learning is even more effective. Finally, Baker and Corbett [6] proposed to contextually estimate whether each student guesses or slips [1]. Their model showed greater accuracy and reliability compared to the conventional BKT model.

While much research leverages performance to assess student knowledge, relatively little research was done using student response time. Beck et.al. modeled student disengagement using student response time [8] and their models were based on the item response theory (IRT). Additionally, Shih B. et al. [12] built a response-time based indicator that can detect good bottom-out hint behaviors (i.e., exploit hints as worked examples). Finally, Wang and Heffernan [14] combined the BKT model together with student response time to predict student performance and their results showed that the proposed model was slightly better than using the BKT alone. Note that they did not incorporate response time into the BKT model while we directly incorporated response time within the Bayesian framework and explored three types of

UMAP '16, July 13–17, 2016, Halifax, Nova Scotia, Canada.

ISBN 978-1-4503-4370-1/16/07...$15.00

DOI: http://dx.doi.org/10.1145/2930238.2930291

observation: *performance*, *time* and *combined* to model student knowledge.

Finally, prior research on student modeling showed that it is still an open question whether incorporating various instructional interventions into the BKT would indeed lead to better performance. For example, Beck et al. proposed the HELP model [2] to measure the impact of the tutors' help. Their results showed that HELP model did not yield a more accurate prediction compared to the BKT.

3. METHOD

Fundamentally, the BKT model is a two-state Hidden Markov Model (HMM). It is [4] characterized by five parameters *Prior Knowledge*, *Learning rate*, *Forget Rate*, *Guess* and *Slip*. The BKT [5] model continually updates its parameters based upon the observation of student's performance history. Note that, the conventional BKT does not take the different types of instructional interventions into account.

Intervention-BKT is a special case of Input Output Hidden Markov Model (IOHMM) [4], which is extended from HMM. Therefore, fundamentally the Intervention-BKT model is an extension to the BKT model. The Bayesian network topology of the Intervention-BKT is displayed in Figure 1. Compared with BKT, Intervention-BKT adds a sequence of unshaded input nodes I. The input nodes I represent instructional interventions. Each input node I carries a pair of extra edges with arrows pointing to the corresponding knowledge state S and student observation nodes O. The arrows between input nodes I and student observation nodes O represent how instructional interventions affect a student's performance. The arrows between input nodes I and knowledge state nodes S represent how instructional interventions affect a student's hidden knowledge state. Thus, the Intervention-BKT employs $1 + 4 \times K$ parameters (compared with 5 parameters of BKT). The Prior Knowledge share the same definition as the conventional BKT: Prior Knowledge= P(S_0=learned). For each of the K types of interventions $A_j, j \in [1, K]$, the Intervention-BKT defines four conditional parameters, the *Learning rate*, *Forget Rate*, *Guess* and *Slip* parameters for A_j :

$$\begin{aligned}
&\textbf{Learning Rate}_{\mathbf{A_j}} = \text{P(learned}|\text{unlearned, } \mathbf{I_t} = \mathbf{A_j}\text{)}\\
&\textbf{Forget}_{\mathbf{A_j}} = \text{P(unlearned}|\text{learned, } \mathbf{I_t} = \mathbf{A_j}\text{)}\\
&\textbf{Guess}_{\mathbf{A_j}} = \text{P(correct}|\text{unlearned, } \mathbf{I_t} = \mathbf{A_j}\text{)}\\
&\textbf{Slip}_{\mathbf{A_j}} = \text{P(incorrect}|\text{learned, } \mathbf{I_t} = \mathbf{A_j}\text{)}
\end{aligned}$$

In this paper, we mainly focus on modeling two types of instructional intervention (K=2) *elicit* and *tell*. A possible sequence of instructional interventions is suggested above input node in Figure 1. Note that the Intervention-BKT model is trained from a sequence of instructional interventions and a sequence of corresponding performance extracted from the log files directly.

The second goal of our paper is to explore the benefits of using student response time to infer student hidden knowledge, instead of using students' performance alone. Response time measures how long a student has spent on a given attempt. It is denoted by one of two symbols: *quick* and *slow*. The symbols were assigned by comparing the student's response time on that given step with the median response time of all students on the same step. If the time is greater than the median, we classify it as *slow*, otherwise, *quick*. Note that, when tutor *tells*, we automatically assigned the symbol *quick* based on the assumption that all students spent the same amount of time reading the content.

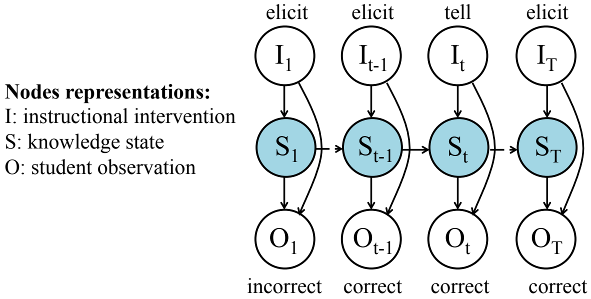

Figure 1: The Bayesian network topology of the Intervention-BKT model

In short, we constructed both the BKT and the Interventional-BKT with three types of observations: 1) *performance*: the correctness of entry on a step; 2) *time*: the speed of the student's response on a step and 3) *combined*: a combination of *performance* and *time*.

4. TRAINING CORPUS

Cordillera is a Natural Language ITS teaching college level introductory physics. All participants in our training corpus experienced identical procedure: 1) completed a survey; 2) read a textbook; 3) took a pretest; 4) solved seven training problems, and finally 5) took a post-test.

In the learning literature, it is commonly considered that relevant knowledge in domains such as math and science is structured as a set of independent but co-occurring Knowledge Components (KCs). A *Knowledge Component (KC)* is the atomic unit of knowledge. It is: "a generalization of everyday terms like concept, principle, fact, or skill, and cognitive science terms like schema, production rule, misconception, or facet" . It is assumed that the student's knowledge state at one KC has no impact on the student's understanding of any other KCs. This is an idealization, but it has served ITS developers well for many decades, and is a fundamental assumption made by many student models [5].

Cordillera consists of a subset of the physics work-energy domain, which is characterized by five primary KCs: Kinetic Energy(KE), Gravitational Potential Energy(GPE), Spring Potential Energy (SPE), Total Mechanical Energy (TME) and Conservation of Total Mechanical Energy (CTME). Given the KCs' independence assumptions, our student model was constructed and evaluated for each of the five primary KCs individually. However, in Cordillera, some steps have mixed KC, thus we also trained on sequences of observations irregardless of the KCs involved (denoted by OVERALL).

Cordillera provides two types of instructional interventions *elicit* and *tell*. *Elicit* usually takes the form of a question, e.g., which principle will help you calculate the rock's instantaneous magnitude of velocity at T1? *Tell* usually takes the form of a written statement, i.e., to calculate the rock's instantaneous magnitude of velocity at T1, we will apply the definition of kinetic energy again.

In our datasets, the instructional intervention *elicit* or *tell* were guided by different pedagogical rules. We used

Table 1: Accuracy in Next Step Performance Prediction

KC	**Data**		**BKT**			**BKT (without tell)**			**Intervention-BKT**		
			Perf	**Time**	**Comb**	**Perf**	**Time**	**Comb**	**Perf**	**Time**	**Comb**
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
	Eff	**1**	0.814*	0.636	0.814*	0.787	0.484	0.805	0.814*	0.535	0.751
K	Acr	**2**	0.740	0.652	0.740	0.734	0.496	0.740	0.749*	0.630	0.746
	Ine	**3**	0.717	0.682	0.717	0.716	0.512	0.708	0.728*	0.670	0.728*
	Eff	**4**	0.766	0.557	0.766	0.772*	0.530	0.757	0.763	0.602	0.763
G	Acr	**5**	0.704*	0.643	0.704*	0.704*	0.532	0.704*	0.696	0.608	0.692
	Ine	**6**	0.685*	0.668	0.685*	0.685*	0.530	0.685*	0.683	0.611	0.679
	Eff	**7**	0.825*	0.779	0.825*	0.825*	0.318	0.825*	0.825*	0.782	0.820
S	Acr	**8**	0.713	0.713	0.713	0.713	0.440	0.713	0.720*	0.713	0.720*
	Ine	**9**	0.680	0.680	0.680	0.649	0.474	0.660	0.700*	0.680	0.700*
	Eff	**10**	0.778*	0.599	0.778*	0.762	0.466	0.778*	0.777	0.538	0.675
T	Acr	**11**	0.689	0.593	0.689	0.693	0.480	0.689	0.701*	0.598	0.694
	Ine	**12**	0.660	0.601	0.660	0.667	0.486	0.660	0.675*	0.628	0.675*
	Eff	**13**	0.771	0.771	0.771	0.771	0.628	0.771	0.734	0.775*	0.725
C	Acr	**14**	0.657	0.657	0.657	0.650	0.593	0.650	0.665*	0.629	0.650
	Ine	**15**	0.635	0.635	0.635	0.555	0.584	0.628	0.650*	0.604	0.626
	Eff	**16**	0.790	0.659	0.785	0.782	0.507	0.787	0.794*	0.542	0.786
O	Acr	**17**	0.708	0.650	0.708	0.699	0.511	0.708	0.722*	0.598	0.712
	Ine	**18**	0.683	0.663	0.683	0.678	0.521	0.683	0.698*	0.604	0.694

Note: the highest accuracy are marked by * and best models are shaded

three types of training datasets *Effective*, *Ineffective* and *Across*. According to prior literature [3], *Effective* was generated from training corpus implementing effective pedagogical rules; while the *Ineffective* datasets was generated from training corpus implementing ineffective rules contributing less to student learning. *Across* contained both datasets. Since students learn differently in each training corpus, we trained a separate model for each of them.

The overall dataset comprises 38028 data points from 158 students. Among them, 5810 data points from 29 students belong to the *Effective* dataset and 32218 data points from 129 students belong to the *Ineffective* dataset. There were no significant training time difference among these three datasets. On average, it took students roughly 4-9 hours to complete the training. The average number of Cordillera-student interactions was more than 280. A data point in our training datasets is either the first attempt by a student in response to a system *elicit*, or a system *tell* during the student's training on Cordillera.

5. EXPERIMENT

In our experiment, nine models {BKT, BKT (without tell), Intervention-BKT} × {performance, time, combined} were evaluated across three corpus {*Effective*, *Across*, *Ineffective*} on six primary KCs {KE, GPE, SPE, TME, CTME, OVERALL}. Thus, we constructed nine models for the eighteen datasets. Note that BKT (without tell) only considers the student observation corresponding to a tutor elicits, while BKT considers the correctness in a log file irregardless of whether it is generated by tutor *elicits* or tutor *tells*.

We focused on two prediction tasks. The first task is to predict students' next step performance in training, referred to as "next step performance predictions". The second task is to predict their post-test scores, referred to as "post-test scores predictions". For the first task, the model estimates $P(S_t = learned)$ at each learning opportunity, then uses the previous state probability to predict the next observation $P(O_t = correct)$ by using the formula below. The formula for conventional BKT is shown in Equation (1) and (2).

$$P(S_t = learned) \tag{1}$$
$$= P(S_{t-1} = learned) * (1 - Forget) + P(S_{t-1} = unlearned) * Learning\ Rate$$

$$P(O_t = correct) \tag{2}$$
$$= P(S_t = learned) * (1 - Slip) + P(S_t = unlearned) * Guess$$

For Intervention-BKT, it uses a similar equations but with *Learning Rate*, *Forget Rate*, *Guess* and *Slip* parameters conditioned on the intervention type at time t, that is I_t.

Note that for different types of observations, *Slip* and *Guess* were calculated differently. When we used *performance*, *Slip* and *Guess* have already been learned and their values can be looked up in the emission probability table. When we used *time*, the model could not estimate *Slip* and *Guess*, as it only learns the probability that, given the students' learning states, whether their observation of response time is *quick* or *slow*. Therefore, we set both *Slip* and *Guess* to be 0.2 based on expert guess. When we used *combined*, we calculated *Slip* and *Guess* by looking up the values of the entries in emission tables. And calculate the *Guess* and *Slip* using the formula as shown below for BKT. Similarly, for Intervention-BKT, *Guess* and *Slip* are conditioned on the corresponding instructional interventions.

$$Slip = P(incorrect, slow|learned) \tag{3}$$
$$+ P(incorrect, quick|learned)$$

$$Guess = P(correct, slow|unlearned) \tag{4}$$
$$+ P(correct, quick|unlearned)$$

For the second task, both BKT and Intervention-BKT

Table 2: RMSE in Post-test Score Prediction

KC	Data		BKT			BKT (without tell)			Intervention-BKT		
			Perf	**Time**	**Comb**	**Perf**	**Time**	**Comb**	**Perf**	**Time**	**Comb**
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
K	Eff	**1**	0.295	0.236	0.288	0.279	0.298	0.214*	0.271	0.260	0.235
	Acr	**2**	0.357	0.235	0.359	0.263	0.242	0.241	0.249	0.190*	0.244
	Ine	**3**	0.378	0.234	0.380	0.261	0.222	0.245	0.246	0.183*	0.242
G	Eff	**4**	0.229	0.165	0.226	0.160*	0.285	0.167	0.175	0.248	0.176
	Acr	**5**	0.306	0.263	0.306	0.212*	0.306	0.218	0.233	0.267	0.229
	Ine	**6**	0.330	0.276	0.330	0.261	0.222	0.245	0.246	0.183*	0.242
S	Eff	**7**	0.368	0.254	0.367	0.283	0.290	0.288	0.317	0.215*	0.316
	Acr	**8**	0.420	0.319	0.418	0.278	0.296	0.279	0.289	0.264*	0.286
	Ine	**9**	0.434	0.335	0.432	0.278*	0.300	0.279	0.286	0.279	0.284
T	Eff	**10**	0.287	0.254	0.277	0.233	0.281	0.194*	0.218	0.232	0.196
	Acr	**11**	0.347	0.233	0.343	0.236	0.282	0.204	0.229	0.198*	0.220
	Ine	**12**	0.363	0.236	0.355	0.231	0.277	0.211	0.229	0.194*	0.222
C	Eff	**13**	0.307	0.216*	0.296	0.246	0.263	0.244	0.256	0.258	0.254
	Acr	**14**	0.326	0.278	0.323	0.244*	0.319	0.246	0.254	0.295	0.252
	Ine	**15**	0.336	0.293	0.334	0.244*	0.330	0.250	0.257	0.303	0.256
O	Eff	**16**	0.332	0.180*	0.331	0.284	0.256	0.234	0.268	0.220	0.249
	Acr	**17**	0.367	0.249	0.367	0.256	0.270	0.240*	0.277	0.244	0.262
	Ine	**18**	0.381	0.269	0.382	0.252	0.274	0.242*	0.279	0.253	0.266

Note: the lowest RMSE are marked by * and best models are shaded

traced a student's knowledge until the last step on Cordillera. Then the probability that a student gives a correct response in post-test can be calculated based on the probability that the student is in the learned state on the final step. We assume that no learning occurs during the post-test and thus the students' knowledge state would not change.

6. RESULTS

Accuracy and Leave One Out Cross Validation Root Mean Square Error (LOOCV RMSE) were used to evaluate the outcomes of these two prediction tasks respectively. Accuracy evaluates how well our models correctly identify an *incorrect* or a *correct* student responses: the higher the value, the better. LOOCV RMSE measures the difference between our predicted post-test scores and the actual post-test scores: the lower the value, the better.

Table 1 shows the model's **accuracy** in predicting students' next step performance during training. Columns represent 9 different models. Rows represent 18 different datsets: 6 primary KCs (denoted by K, G, S, T, C and O) on 3 types of training corpus (denoted by Eff, Acr and Ine). The highest accuracy among the 9 models for each dataset is marked *. Then the best model is selected among the ones producing the highest accuracy and when there is a tie, the best model is the one *involving the least number of parameters*. The cells contain the best model are shaded.

Table 1 shows that 12 out of the 18 shaded cells are produced by Intervention-BKT. 5 of them are BKT (without tell) and only 1 of them is BKT. Thus, the Intervention-BKT seemingly leads to better prediction than BKT-based models. Moreover, among the three types of observations, performance is the best choice for next step performance prediction since 16 out of 18 best models used performance.

Table 2 presents the **LOOCV RMSE** for Post-test Scores Predictions. For this task, the lowest LOOCV RMSE are all marked * and the best models are shaded. From the shaded cells we can see there is no clear winner between BKT (without tell) and Interventional-BKT as they generate 9 and 7 best models respectively. The conventional BKT produces the worst result. 13 out of 18 best models use *time* or *combined* observations while only 5 are produced by using *performance*. It seems including students' response time makes better predictions than using *performance* only.

7. DISCUSSION

In this paper, we made two major contributions: 1) we leveraged student response time to infer students' knowledge, and 2) we proposed Intervention-BKT that can incorporate different types of instructional interventions into student models and learn different parameters for each type of interventions. We trained nine different model variations and tested them on two types of predictions: 1) students' next step performance prediction and 2) students' post-test scores predictions.

For future work, we will explore other *Guess* and *Slip* parameter when using *time* instead of using 0.2 based on expert guess. Secondly, we will explore other ways to classify students' response time: for example, classifying them into {*too fast*, *reasonable*, *too slow*} instead of {*quick*, *slow*}. Third, we will evaluate the effectiveness of our models to other datasets in other domains to determine whether our proposed model is indeed robust. Finally, we will apply our model in systems that involve other types of tutor instructional interventions, such as *skip* (*elicit* a question without asking students for explanation) and *justify* (ask students to explain after they give an answer).

Acknowledgments

This research was supported by the NSF Grant 1432156 ”Educational Data Mining for Individualized Instruction in STEM Learning Environments”.

8. REFERENCES

[1] J. Beck. Difficulties in inferring student knowledge from observations (and why you should care). In *Educational Data Mining: Supplementary Proceedings of the 13th International Conference of Artificial Intelligence in Education*, pages 21–30, 2007.

[2] J. E. Beck, K.-m. Chang, J. Mostow, and A. Corbett. Does help help? introducing the bayesian evaluation and assessment methodology. In *ITS*, pages 383–394. Springer, 2008.

[3] M. Chi, K. VanLehn, and D. Litman. Do micro-level tutorial decisions matter: Applying reinforcement learning to induce pedagogical tutorial tactics. In *Intelligent Tutoring Systems*, pages 224–234. Springer, 2010.

[4] S. Chiappa and S. Bengio. Hmm and iohmm modeling of eeg rhythms for asynchronous bci systems. Technical report, IDIAP, 2003.

[5] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *UMAP*, 4(4):253–278, 1994.

[6] R. S. d Baker, A. T. Corbett, and V. Aleven. More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In *Intelligent Tutoring Systems*, pages 406–415. Springer, 2008.

[7] W. J. González-Espada and D. W. Bullock. Innovative applications of classroom response systems: Investigating students' item response times in relation to final course grade, gender, general point average, and high school act scores. *Electronic Journal for the Integration of Technology in Education*, 6:97–108.

[8] E. Joseph. Engagement tracing: using response times to model student disengagement. *Artificial intelligence in education: Supporting learning through intelligent and socially informed technology*, 125:88, 2005.

[9] Z. A. Pardos and N. T. Heffernan. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *UMAP*, pages 255–266. Springer, 2010.

[10] Z. A. Pardos and N. T. Heffernan. Kt-idem: Introducing item difficulty to the knowledge tracing model. In *User Modeling, Adaption and Personalization*, pages 243–254. Springer, 2011.

[11] D. L. Schnipke and D. J. Scrams. Exploring issues of examinee behavior: Insights gained from response-time analyses. *Computer-based testing: Building the foundation for future assessments*, pages 237–266, 2002.

[12] B. Shih, K. R. Koedinger, and R. Scheines. A response time model for bottom-out hints as worked examples. *Handbook of educational data mining*, pages 201–212, 2011.

[13] R. D. L. V. S. Thomas et al. *Response Times: Their Role in Inferring Elementary Mental Organization: Their Role in Inferring Elementary Mental Organization*. Oxford University Press, USA, 1986.

[14] Y. Wang and N. T. Heffernan. Leveraging first response time into the knowledge tracing model. *International Educational Data Mining Society*, 2012.

[15] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon. Individualized bayesian knowledge tracing models. In *Artificial intelligence in education*, pages 171–180. Springer, 2013.