UNIVERSITY OF PITTSBURGH

INDEPENDENT STUDY - SPRING 2018

Dynamic Knowledge Modeling with Heterogeneous Activities for Adaptive Textbooks

---

Author:
Khushboo THAKER

Supervisor:
Dr. Peter BRUSILOVKSY

A report submitted in fulfillment of the requirements for Independent study

in the

PAWS LAB
Intelligent Systems Program, School of Computing and Information

UNIVERSITY OF PITTSBURGH

Abstract

Intelligent Systems Program, School of Computing and Information

ISSP 2990

Dynamic Knowledge Modeling with Heterogeneous Activities for Adaptive Textbooks

by Khushboo THAKER

Online textbooks produce large amount of data related to student interactions. *Adaptive textbooks* attempt to use this interaction data to infer the current state of student knowledge and recommend most relevant learning materials. A challenge of student modeling in the context of adaptive textbooks is that traditional student models are constructed based on performance data (question answers or problem solving). Student interaction with online textbooks, however, produces large volume of student reading data, but a very limited amount of question-answering data. In this work, we propose a dynamic student knowledge modeling framework for online adaptive textbooks, which utilizes student reading data combined with few available quiz activities to better infer the current state of knowledge. We evaluate our models on the dataset collected from an Information Retrieval course at the University in Pittsburgh. Results show that our model can predict future student reading time and quiz performance significantly better than traditional Knowledge Tracing methods. This framework serves as a preliminary effort demonstrating the importance and feasibility of combining heterogeneous activities for dynamic knowledge modeling in the textbook learning context.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Adaptive online textbooks could be considered as one of the oldest technologies of personalized Web-based learning [Hen+99; WB01; Kav04]. A gradual shift to electronic books and textbooks over the last 10 years makes this technology even more attractive than in its early days. The challenge for the modern research on adaptive textbooks is its integration with other online learning tools - problems, questions, animations, etc. In particular, student modeling approaches based on textbook readings behavior should be made compatible with more traditional student modeling based on student performance. This compatibility would support important “cross-content” recommendation where pages to read could be recommended through the analysis of problem-solving performance while interactive content (animations, problems, questions) could be recommended by considering the reading progress.

In performance-oriented intelligent tutoring systems (ITS) student knowledge state is measured on the level of individual skills or concepts of study. These knowledge units are known as Knowledge Components (KCs). Main goal of KC-level knowledge modeling is to provide effective learning and reduce the total time of skill acquisition of a student, as the system could guide the student to the most appropriate learning content. To support this personalization, the system keeps track of students performance such as problem-solving and question-answering. These user interactions are later used by student modeling systems to distill student knowledge and predict student behavior.

Unfortunately, this well-explored approach could not be directly applied for adaptive textbooks. In most of the cases, textbook reading logs provide only a small fraction of performance data (data on question answering and other activities related to course), which is not sufficient for timely and reliable student modeling. Naturally, these reading logs provide a huge amount of data on student reading and interaction with text, however, the use of this data for student modeling is not straightforward:

- The reading logs are noisy and not accurate. For example, a student can open a course content, start reading and then leaves for some personal work, as the system is open, till time out the system will log as student was reading that content.
- Individual differences of students based on different reading proficiency, different background and study goal

In this paper, we present and evaluate a novel approach that combines heterogeneous student activities (both reading data and performance data) to construct dynamic student knowledge model for adaptive textbooks. The rest of this paper is organized as follows: Section 2 discusses several directions of related work; Section 3 describes the proposed approach; Section 4 introduces the evaluation setup; Section 5 presents and discusses experimental results; Section 6 summarizes conclusions and directions of future work.

Chapter 2

Related Work

2.0.1 Student Modeling in ITS

Approaches in student modeling in ITS could be classified into two major groups: Logistic Regression models and Knowledge Tracing models [Pel17]. Logistic regression models are motivated by the power law of learning [NR81], which states that probability of applying a skill correctly decreases by a power function. These models utilize student observation logs as the inputs, and try to predict student performance with a learning activity based on KCs (skills) associated with the activity. One of the basic models in this group is known as Additive Factor Model (AFM) [CKJ06], which computes the odds of a student's success on a particular question based on the number of previous attempts. Performance Factor Analysis [PCK09] improves AFM by separately modeling the student's previous successes and failures on a particular skill. Although logistic regression models are efficient in predicting students future performance, they lack the ability to represent knowledge estimates for each individual KC, which is important for personalization in adaptive systems. In contrast, Knowledge Tracing (KT) models [CA95] directly represent KC-level knowledge estimation and allow dynamic knowledge update. Knowledge Tracing uses Hidden Markov Models (HMM) to model student knowledge as binary latent variables. Each latent variable represent student knowledge of a particular KC, which could be either known or unknown. The observed variable is the performance of student at a given step, which is usually measured as a binary representing correctness of a step or an answer (correct or not correct). The emission probability in KT is defined as the probability of *guess* or *slip* and transmission probability is defined as transfer from unknown state to known state. In our work presented in this paper, we follow the KT modeling approach since we need knowledge estimates of different KCs to support several kinds of personalization.

2.0.2 Adaptive Online Textbooks

The research on adaptive textbooks has been motivated by the increased popularity of WWW and the opportunity to use this platform for learning. The hypertext nature of early WWW made an online hypertext-based textbook a natural media for learning while the increased diversity of Web users stressed the need for adaptation. The first generation of adaptive textbooks [De 97; BE98; Hen+99; Kav04] focused on tracing student reading behavior to guide students to most relevant pages using adaptive navigation support [De 97; BE98; Hen+99; WB01] or recommendation [Kav04]. These types of personalization were based on a sophisticated knowledge modeling: each textbook page was associated with a set of concepts *presented* on the page as well as concepts *required* to understand the page [De 97; BE98]. On the other hand, student modeling was relatively simple: these systems treated each visit to a page as a contribution to learning all presented concepts.

An important trend of modern online textbooks is the increased inclusion of interactive content "beyond text". While the attempts to integrate online reading with problem solving have been made in the early days of online textbooks [WB01], it was a rare exception. Modern textbooks, however, routinely integrate a variety of "smart content" such as visualizations [Röß+06], problems [Eri+15], and videos [KKK14]. In this context, the ability to integrate data about student work with all these components and use it for a better-quality student modeling becomes a challenge for modern online textbooks.

Chapter 3

Knowledge Modeling in textbooks

Our work attempts to combine the ideas of reading-based student modeling explored in the area of adaptive textbooks with the ideas of performance based modeling explored by traditional ITS. Our goal is to develop more reliable modeling for modern adaptive textbooks that could support several kinds of personalization such as guiding students to most appropriate sections or recommending relevant external content. This section introduce our earlier work on student modeling in textbooks and presents two novel models that combine reading-based KT [Hua+16] with traditional KT [CA95] thus leveraging both reading and question-answering data.

3.0.1 Behavior Model (BM) and Its Problems

As a baseline model in this work we use Behavior Model suggested and explored earlier by our group [Hua+16]. To build BM, we modified traditional knowledge tracing to infer knowledge from students reading behavior as shown in Figure 3.1. Each reading activity performed by a student is labeled as either *Read* or *Skim*. *Read* labeled observation declares that student was reading the material within average reading speed threshold, while *Skim* on other hand declares that student skimmed the material. Following the practice of adaptive textbooks, each document in the textbook is mapped to the corresponding KCs. The assigned reading speed label on a document is assigned to KCs mapped to the document. The observation node is a binary reading variable (*Read* or *Skim*) and the hidden node is a binary knowledge state variable (*Learned* and *Unlearned*). The interpretation of parameters of KT in this context are as below:

- **P(L_0):** the probability that a student initially knows the KC, i.e., the student is in the learned state.
- **P(T):** the probability that a student transitions from an unlearned to a learned state.
- **P(G):** the probability of a student to *Skim* when being in the unlearned state.
- **P(S):** the probability of a student to *Read* when being in the learned state.

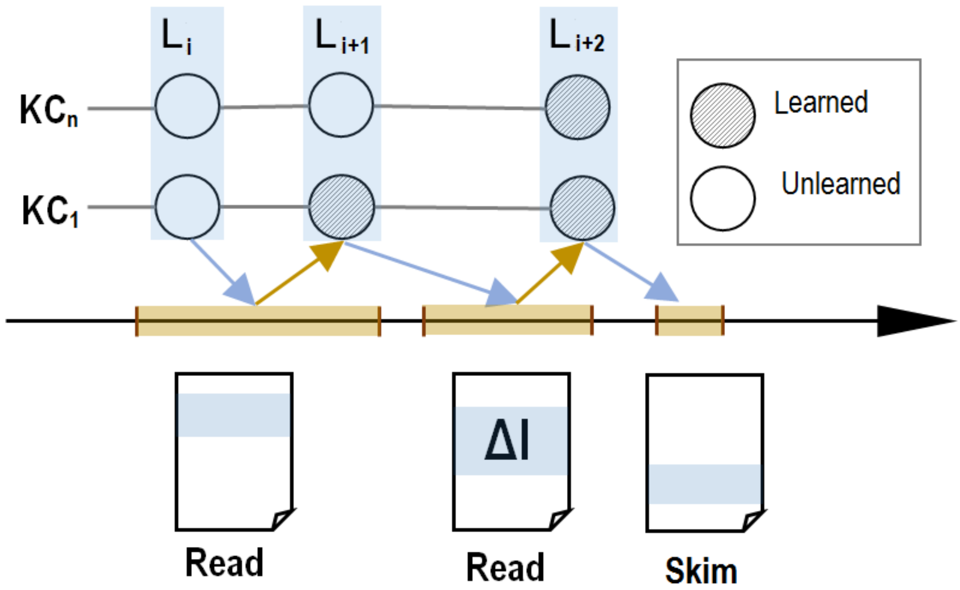
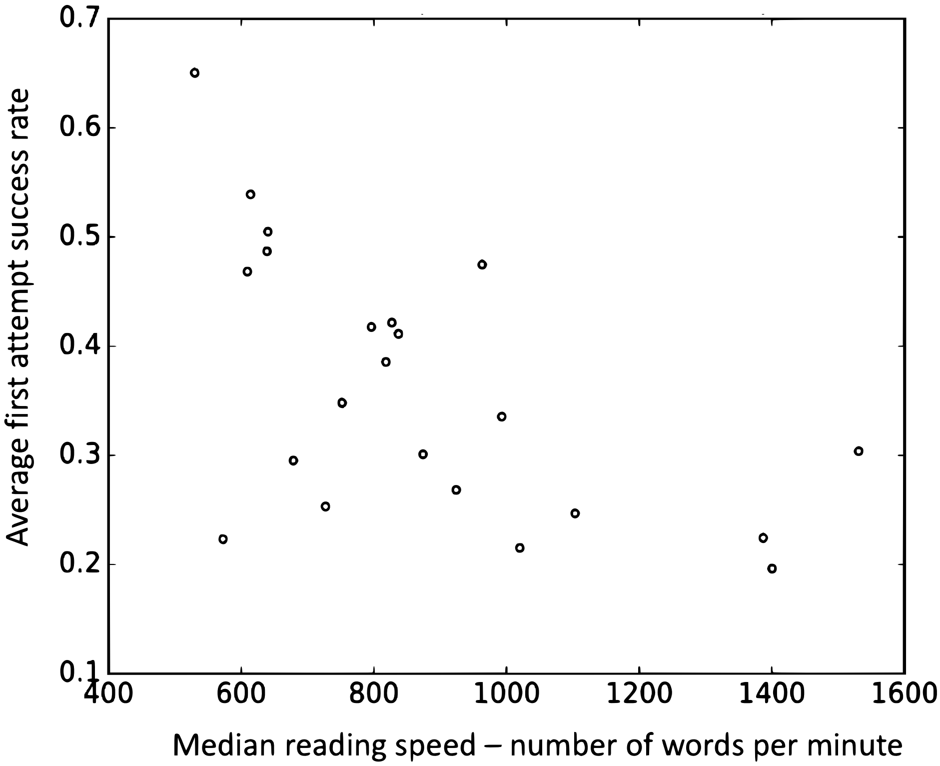


FIGURE 3.1: An illustration of knowledge modeling in textbook reading. KC_i are knowledge components L_i denotes the students' knowledge state for each KC at i^{th} learning opportunity. ΔI indicates the content that a student reads.

The BM model has a strict assumption that students reading speed is positively correlated with their knowledge state. However, other research indicated that this this assumption might not always hold [Bak+04]. Indeed, in the dataset we considered for this study we observed the general negative correlation between student reading behavior and quiz performance as shown in Figure 3.2. Pearson correlation of -0.58 is an indicator of data consists of mixture different types of students and noisy reading interactions. As can be seen from the Figure 3.2 a different subpopulation of students consists of (1) students with high reading speed and low performance (likely, students gaming the system); (2) students who read slow and have high performance; (3) students that read slow and have low performance.

FIGURE 3.2: Correlation between students reading speed and average first attempt success rate (Pearson correlation is −0.58)

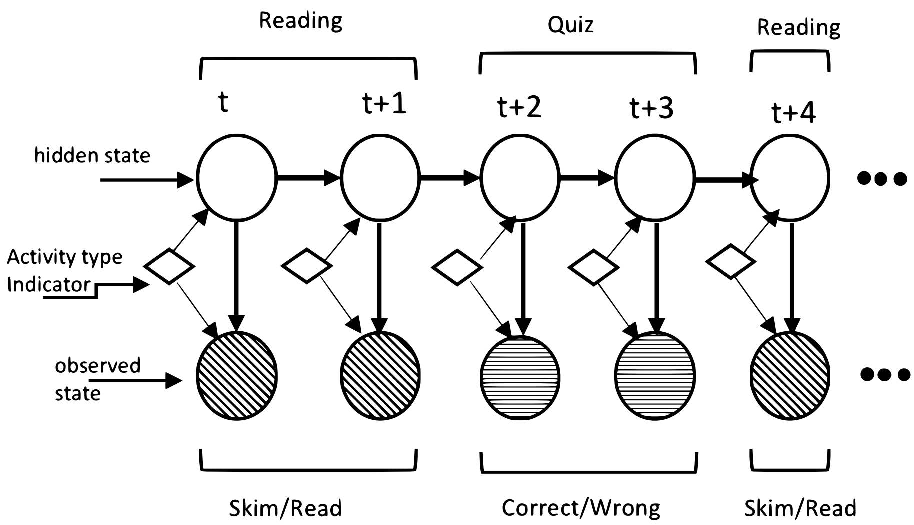


The primary goal of the work presented in this paper was to improve the simplistic BM. Our key ideas are (1) to handle mixture and noisy reading behavior among students by tuning it with other available activities performed by the student in the online textbook-based learning environment and (2) incorporate individual student differences to address better knowledge estimation for different types of students. In two following subsections we present two models that advance the original BM model in the proposed directions.

3.0.2 Behavior-Performance Model (BPM)

We propose *Behavior-Performance Model (BPM)* that integrates different types of activities (reading speed and quiz performance) in a basic HMM framework (Figure 3.3). There are two key characteristics of our model. Firstly, different from training separate HMMs for different activity types, our model shares latent knowledge states directly across different activity types within the same HMM for each KC. The quiz performance activities could thus be directly utilized to conduct Bayesian update for tuning the knowledge inferred from reading behavior activities. Secondly, our model fits two sets of parameters within one HMM per KC, one set of reading activity part, one set for quiz activity part. It is important to get a different set of parameters for quiz part because comparing with noisy reading behavior, quiz (first attempt) performance should have high positive correlation with knowledge state, and thus should have much lower guess and slip parameters (Section 5.0.3 feature analysis confirms). To achieve this we utilized Feature Aware Student Knowledge Tracing (FAST) framework [Kha+14], which replaces the conditional probability tables of the emission and transmission probabilities in KT framework with logistic regression distribution. HMM parameters are thus computed based on logistic regression models with features at each time step. This allows flexibility of incorporating a large number of features in logistic regression components of the model. To enable FAST for different types of observation variables we introduce an activity type indicator variable which sets 0 for reading activities and 1 for quiz activities for each logistic regression component of the parameters for each KC (see Figure 3.3).

FIGURE 3.3: Behavior Performance Model (BPM)



3.0.3 Individualized Behavior-Performance Model (IBPM)

In *BPM* model we incorporate reading activities as binary variables with values *Skim* and *Read*. Reading speed being a continuous variable, discretization of this manner causes a lot of information loss at student level. This information might be very helpful to characterize individualized student reading behavior and to obtain individualized parameters for different kinds of students. We propose *Individualized Behavior-Performance Model (IBPM)* that incorporates the individualized reading speed information as a feature in addition to activity type indicator features. This feature is based on accumulated median reading speed from first reading activity till $(i-1)$th reading activity of a student, where i is the current step of observation in an HMM of a KC as shown in Figure 3.4. The feature is normalized to be in the range of 0 to 1 as there is a large variance in reading speed observation. Thus at each step along with different activity sequence observed, the model is also provided individual average reading speed observed so far. There are several benefits of our method:

- This method provides different sets of parameters (learn, guess, slip) for students with different reading speed.
- Compared with adding a student dummy variable per-student for individualization, this feature generalize across students, because it learns the general association of the speed with HMM parameters within a KC.
- It is a flexible approach to integrate other behavior features as FAST has linear complexity in respect to the number of features.

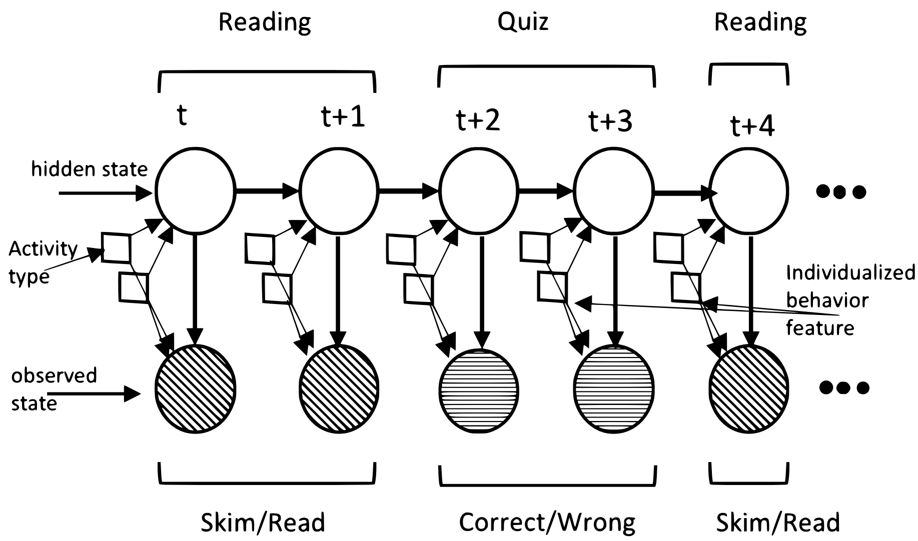


FIGURE 3.4: Individualized Behavior Performance Model (IBPM)

Chapter 4

Experiments

4.0.1 System and Dataset

The dataset used for the experiment is collected from online reading platform Reading Circle [GPB13] in Spring 2016. This system was used for graduate level course on Information Retrieval at University in North America. The system provides an active reading environment to the student where they read the assigned textbooks material to prepare for the next class. To keep students motivated to use the system for reading, the system provides feedback about students reading progress as well as average class reading progress. Each section of the assigned textbook reading is followed by a quiz with several questions, which allow students to assess how well they learned the content. There is no restriction on the number of attempts to the questions, Reading Circle logs each and every attempt made by the student. The final dataset contains 22,536 interactions from 22 students (see more details in Table 4.1).

TABLE 4.1: Dataset Statistics

Number of documents (sections)	394
Number of questions	158
Number of students	22
Median per student of reading time (minutes)	104
Average per student questions attempted	126
Median Reading Speed (words per minutes)	773
Percentage of skimming Activities	33%
Percentage of reading Activities	67%

4.0.2 Data-Preprocessing

Reading Speed Discretization

Discretization of reading time is essential for labeling the observations to *Read* and *Skim*. For discretization we followed the same technique as performed by Huang et. al. [Hua+16] As this discretization doesn't consider the student individual differences, we conducted a study where we tried to incorporate individual reading speed as a feature for our knowledge tracing framework (see details in Section 3.0.3).

Knowledge Component Extraction

The key to well trained Bayesian Knowledge Tracing framework is to have correct representative KCs. KCs work as knowledge units on which students knowledge is inferred. Traditional way of defining KCs is manual knowledge modeling by subject experts. Recently, Huang et al.[Hua+16], tried different KC extraction methods and found automatic word-based method to be reliable. However, word-based method gives a large set of KCs in our case (9000+) and it is very noisy. To improve automatic KC extraction based on words' importance in a reading unit, we applied the traditional TF*IDF (Term Frequency - Inverse Document Frequency) approach [SB88]. For each document (reading unit) top 5 TF*IDF-weighed words were extracted and considered as KCs for that document. This reduced the number of KCs to 630. Note that before TF*IDF weighting and KC extraction, each document is tokenized by stop-word removal, excluding non-letter symbols (e.g. punctuation marks and digits) and finally, and stemming by Porter stemmer [Por80] .

Tools and Parameters

For building both the variation of knowledge tracing models, we used open source FAST [GHB14] toolkit. HMM models are prone to get trained for local optimum values, due to which proper initialization of HMM parameters is very important. In all the models the HMM modes were initialized with (0.1,0.1,0.8,0.8) parameter values for $P(L_0)$, $P(T)$, $P(G)$ and $P(S)$ respectively. This choice of initialization is based on observing the negative correlation in our exploratory data analysis (Figure 3.2) and our preliminary experiments where the predictive performance of each baseline and proposed model under another initial parameter set (0.1, 0.1, 0.2, 0.2) was worse than using this set.

Baseline Methods

In order to show the performance gain of our approach we used two variation of Bayesian Knowledge Tracing as baselines. First model is the *Behavior Model* (**BM**)) reviewed in section 3.0.1. It is trained using reading interactions and provides a baseline for reading time prediction. Second model is traditional ITS *Performance Model* (**PM**) trained on question answers performed by the student. In addition we use a majority class baseline (**MC**). As the proposed model is able to perform both reading time prediction as well as quiz performance prediction, this choice of baseline models separately act as baseline for proposed models reading time prediction and quiz performance prediction task.

Cross Validated Prediction Evaluation

FAST trains individual HMM for each KC using training data and performs prediction on test data. Firstly, we randomly selected 50% of students and put all their reading and quiz activity data into training set. Then for the remaining 50% of students, we put the first half of their reading and quiz activity sequence into training set. The second half of their activity sequences are withheld for test set. This process is repeated 10 times. In this way of 10 way split validation, we primarily examine each model's ability to predict students' future (unseen) reading behavior and quiz performance, after utilizing some historical behavior and performance data to establish the initial knowledge estimates. We leave for the future to examine models' performance under different ways of splitting. The prediction are reported on reading speed, first attempt quiz performance and all-attempts quiz performance. For quiz performance prediction, the primary focus is given to first attempt prediction. Since we observed that a subset of students engaged into gaming the system produces a large number of attempts, we regarded first attempt performance as more reliable. 10 split cross validation is performed from the generated folds and Area Under the Receiver Operating Characteristic curve (AUC) and Root Mean Squared Error (RMSE) are reported. We chose to report both RMSE and AUC based on a suggestion from a recent paper, that raised a concern about using only AUC for evaluation of student models [Pel15]. For checking the significance of our results two-sided paired t-test was performed on 10-split results for compared models.

Chapter 5

Results and discussion

5.0.1 Predictive Performance of BPM

Table 5.1 summarizes the predictive performance and Table 5.2 reports statistical test results with Bonferroni correction. Comparing with *MC*, *BPM* has significantly better RMSE and AUC across all prediction tasks except the AUC value on reading speed prediction task. The relatively lower AUC value on reading prediction task indicates a high amount of noise in reading interactions. Since quiz performance usually correlates better with knowledge than reading behaviors, the prediction on quiz is of more importance than that on reading, thus the result in general indicates a clear advantage of *BPM* over *MC*. Comparing with *BM* and *PM* which are trained on a single type of interactions, *BPM* also beats them significantly in corresponding prediction tasks in both RMSE and AUC metrics. We clearly see the advantage of integrating behavior and performance data in *BPM* over traditional models which only utilize a single type of data. Better performance of *BPM* over *BM* indicates that even a small amount of quiz performance data could significantly improve knowledge inference and performance prediction; better performance of *BPM* over *PM* indicates that reading data albeit being noisy, still carries valuable information that could help infer knowledge and conduct prediction.

5.0.2 Predictive Performance of IBPM

The intuition behind *IBPM* is to provide additional student reading behavior features (in addition to activity type indicator) for capturing individual differences better. As can be seen in Table 5.1, *IBPM* incorporating individualized speed feature shows improvement by both RMSE and AUC metrics compared with *BPM*. The improvement is significant for reading speed prediction task and quiz all attempts' performance prediction task. However, its improvement over *BPM* on predicting first attempt performance in terms of RMSE is not significant. A probable reason is that our dataset exhibits a mixture of students in terms of behavior and performance. According to Figure 3.2, we could see that among students with reading speeds lower than the median, there are students with both low and high quiz performance (in terms of first attempts), so the effect of speed on quiz performance might not be captured by a single speed feature. We will construct more individualized features in our future work.

TABLE 5.1: Comparison of performance prediction of reading speed, first attempt quiz prediction (1st att.) and all attempts (all att.) as computed by averaging across 10 splits. The best two results are denoted in bold.

Model	RMSE	AUC	RMSE	AUC	RMSE	AUC
	reading		1st att.		all att.	
IBPM	**0.4832**	**0.5118**	**0.4723**	**0.6351**	**0.3908**	**0.8668**
BPM	**0.4867**	0.4532	**0.4727**	**0.6285**	**0.4135**	**0.8388**
BM	0.4910	0.4388	-	-	-	-
PM	-	-	0.5036	0.6020	0.4265	0.8027
MC	0.5929	**0.5000**	0.5770	0.4890	0.5787	0.5000

TABLE 5.2: Statistical test p value for prediction performance on reading and quiz with Bonferroni correction

Compared Models	RMSE	AUC	RMSE	AUC	RMSE	AUC
	read		1st att.		all att.	
IBPM vs *BPM*	***	***	0.18	*	*	*
IBPM vs *BM/PM*	***	***	***	***	***	***
IBPM vs *MC*	***	**	***	***	***	***
BPM vs *BM/PM*	***	***	*	***	***	*
BPM vs *MC*	***	***	***	***	***	***

10CV paired t-test, p-values
*0.05/5 = 0.01, **0.01/5 = 0.002, ***0.001/5 = 0.0002

5.0.3 Parameter Analysis of BPM

To validate our hypothesis that quiz activities contain less noise than reading activities for inferring knowledge, we conduct a drill-down analysis of parameters of *BPM* and baseline models. We compute parameters for each KC in *BPM* by setting the value of activity type indicator to 0 for the reading part and 1 for quiz part in the logistic regression of each parameter, and then we average the parameters across all KCs. According to Table 5.3, *BPM* has fitted lower *guess* and *slip* parameters in quiz activity part than reading activity part, which indicates that quiz activities have high positive correlation with knowledge state than reading activities i.e., quiz activities indeed have much less noise for inferring knowledge. In addition, Table 5.3 shows that the parameters learned for guess and slip for *BPM* are smaller than those for *BM* and *PM*, which indicates that *BPM* has higher plausibility enabling more accurate knowledge inference than these baseline models [Hua+15].

TABLE 5.3: Parameters learned by different models for learn, guess and slip probabilities

Model	Activity Type	learn	guess	slip
BM	Reading	0.384	0.505	0.776
PM	Quiz	0.091	0.705	0.589
BPM	Reading	0.404	0.363	0.420
BPM	Quiz	0.354	0.288	0.313

Chapter 6

Conclusion and future work

In this paper, we investigated the significance of integrating heterogeneous student activities in knowledge tracing framework for adaptive textbooks. Existing modeling approaches for online textbooks use either reading or performance interactions for dynamic student knowledge modeling. In textbook context when these interactions are noisy, our method of combining different types of interactions can help in tuning the noise and better understand student behavior. To assess this hypothesis, we trained our first model BPM with large volume of noisy reading data and small amount of quiz performance data. The model significantly outperforms the basic models BM which is based on only reading behavior logs, and also significantly outperforms PM which is based on only quiz behavior logs. The results shows that that our hypothesis of addition of quiz interactions will help in learning better student knowledge state is justified.

One limitation of BPM model is the discretization of reading speed observation variable, which causes loss of granular information per student level. To address this, $IBPM$ integrated these continuous observation in student modeling to address student differences. The performance of this model was similar to BPM with a considerable improvement on reading speed prediction and small improvement on quiz performance prediction. One probable reason is the small size of the dataset and the relatively simple features. In the future we would like to construct our models on larger datasets and investigate $IBPM$ further by utilizing other individualization features including average quiz performance, variance in reading speed and considering different subpopulations of students.

Overall, our work could be considered as first attempt to model dynamic student knowledge in textbook context with heterogeneous interactions. We believe that the possibility of integrating individual differences to the proposed model makes it especially promising for real-time learning systems. Moreover, our approach makes it possible to integrate more types of student activities like search, video, listening and discussion to further increase the quality of modeling and to provide holistic student modeling. We plan to explore these opportunities in the future work.

Bibliography

[Bak+04] Ryan Shaun Baker et al. "Off-task Behavior in the Cognitive Tutor Classroom: When Students "Game the System"". In: *Proc. the SIGCHI Conf. on Human Factors in Computing Systems*. CHI '04. Vienna, Austria: ACM, 2004, pp. 383–390.

[BE98] Peter Brusilovsky and John Eklund. "A study of user-model based link annotation in educational hypermedia". In: *Journal of Universal Computer Science* 4.4 (1998), pp. 429–448.

[CA95] Albert T. Corbett and John R. Anderson. "Knowledge tracing: Modelling the acquisition of procedural knowledge". In: *User Modeling and User-Adapted Interaction* 4.4 (1995), pp. 253–278.

[CKJ06] Hao Cen, Kenneth Koedinger, and Brian Junker. "Learning Factors Analysis: A General Method for Cognitive Model Evaluation and Improvement". In: *Intelligent Tutoring Systems*. Vol. 4053. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2006, pp. 164–175.

[De 97] Paul De Bra. "Teaching Through Adaptive Hypertext on the WWW". In: *Int. Journal of Educational Telecommunications* 3.2/3 (1997), pp. 163–180.

[Eri+15] Barbara Ericson et al. "An eBook for Teachers Learning CS Principles". In: *ACM Inroads* 6.4 (2015), pp. 84–86.

[GHB14] José P González-Brenes, Yun Huang, and Peter Brusilovsky. "General features in knowledge tracing: Applications to multiple subskills, temporal item response theory, and expert knowledge". In: *Proc. the 7th Int. Conf. on Educational Data Mining*. 2014, pp. 84–91.

[GPB13] Julio Guerra, Denis Parra, and Peter Brusilovsky. "Encouraging Online Student Reading with Social Visualization". In: *The 2nd Workshop on Intelligent Support for Learning in Groups at the 16th Conf. on Artificial Intelligence in Education*. 2013, pp. 47–50.

[Hen+99] Nicola Henze et al. "Adaptive hyperbooks for constructivist teaching". In: *Kúnstliche Intelligenz* 13.4 (1999), pp. 26–31.

[Hua+15] Yun Huang et al. "A framework for multifaceted evaluation of student models". In: *Proc. the 8th Int. Conf. on Educational Data Mining*. 2015, pp. 203–210.

[Hua+16] Yun Huang et al. "A Framework for Dynamic Knowledge Modeling in Textbook-Based Learning". In: *Proc. 24th Conf. on User Modeling, Adaptation and Personalization*. 2016, pp. 141–150.

[Kav04] A. Kavcic. "Fuzzy User Modeling for Adaptation in Educational Hypermedia". In: *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)* 34.4 (Nov. 2004), pp. 439–449.

[Kha+14] Mohammad Khajah et al. "Integrating Knowledge Tracing and Item Response Theory: A Tale of Two Frameworks". In: *Workshop on Personalization Approaches in Learning Environments (PALE 2014) at the 22th Int. Conf. on User Modeling, Adaptation, and Personalization*. Vol. 1181. CEUR Workshop Proceedings. 2014, pp. 7–12.

[KKK14] Marios Kokkodis, Anitha Kannan, and Krishnaram Kenthapadi. "Assigning Educational Videos at Appropriate Locations in Textbooks". In: *the 7th Int. Conf. on Educational Data Mining (EDM 2014)*. 2014, pp. 201–204.

[NR81] Allen Newell and Paul S Rosenbloom. "Mechanisms of skill acquisition and the law of practice". In: *Cognitive skills and their acquisition* 1 (1981), pp. 1–55.

[PCK09] P.I. Pavlik, H. Cen, and K.R. Koedinger. "Performance Factors Analysis– A New Alternative to Knowledge Tracing". In: *Proc. the 2009 Conf. on Artificial Intelligence in Education: Building Learning Systems that Care: From

Knowledge Representation to Affective Modelling. IOS Press. 2009, pp. 531–538.

[Pel15] Radek Pelánek. “Metrics for evaluation of student models”. In: *Journal of Educational Data Mining* 7.2 (2015), pp. 1–19.

[Pel17] Radek Pelánek. “Bayesian knowledge tracing, logistic models, and beyond: an overview of learner modeling techniques”. In: *User Modeling and User-Adapted Interaction* 27 (2017), pp. 313–350.

[Por80] M.F. Porter. “An algorithm for suffix stripping”. In: *Program* 14.3 (Mar. 1980), pp. 130–137.

[Röß+06] Guido Rößling et al. “Merging Interactive Visualizations with Hypertextbooks and Course Management”. In: *SIGCSE Bull.* 38.4 (2006), pp. 166–181.

[SB88] Gerard Salton and Christopher Buckley. “Term-weighting approaches in automatic text retrieval”. In: *Information processing & management* 24.5 (1988), pp. 513–523.

[WB01] Gerhard Weber and Peter Brusilovsky. “ELM-ART: An adaptive versatile system for Web-based instruction”. In: *Int. Journal of Artificial Intelligence in Education* 12.4 (2001), pp. 351–384.