Modeling Student Learning Outcomes in Studying Programming Language Course

Shanshan Wang, Wenjun Wu
State Key Lab of Software Development Environment
Department of Computer Science and Engineering,
Beihang University
Beijing, China
{wangshanshan, wwj}@nlsde.buaa.edu.cn

Yong Han, Zhenghui Hu
State Key Lab of Software Development Environment
Department of Computer Science and Engineering,
Beihang University
Beijing, China
{hanyong, hzh}@nlsde.buaa.edu.cn

Abstract—Learning outcome assessment is of great significance in the field of traditional on-campus teaching especially on the courses of programming languages. In this work, we take advantage of the data offered by our programming assignment judge system and propose a new IRT-BKT model for estimation of learning outcome. This new framework: Item Response Theory (IRT) model that approximates students’ initial knowledge, and combines it with the estimated difficulty and discrimination of each skill to estimate the probability knew of knowing a skill before practicing it. We then estimate parameters of Bayesian Knowledge Tracing (BKT) probabilities for learn, guess, and slip. Using real data, we show that IRT-BKT model outperforms Item Response Theory and Bayesian Knowledge Tracing in terms of prediction accuracy.

Keywords—Learning Outcome; Programming Language; Item Response Theory; Bayesian Knowledge Tracing

I. INTRODUCTION

Today, information and computer technology is all around us. Programming is not an art accessible to the few and taught at computer science schools anymore, which are taught to wider ordinary students and it have become a crucial tool for any technology related program. As more and more students are taking on programming, it becomes a universal skill, a necessity for every student studying increasingly computerized technology.

Programming as a discipline is difficult to learn and apply successfully, as it has several concepts and structural regulations which have to be understood before something relevant can be achieved. Actually, common consensus is that as a proficient staff in this field takes at least ten years of professional experience [24], and even then the expertise is somewhat limited to a group of certain programming languages and concepts. As for a novice programmer, the learning procedure is usually started by learning the basic structures such as iteration, variables and expressions, which are then extended to array, functions and statements. After learning the structures, the students can combine these “building blocks” to create functional frameworks.

A classical educational system always has a user model – an integral component responsible for keeping track of student progress. The core of a student model is a vocabulary of skills (concepts) that structure the representation of student knowledge. Conceptualizing a set of skills is a hard task in and of itself. However, programming is an inherently structured domain. The basis of a programming language is the grammar that imposes a structure on any code that compiles.

Students’ learning outcome assessment has taken a crucial part in field of education. Accurate outcome assessment not only gives students a more equitable course score, but also assist teachers in finding difficulties and doubts in student mastery of knowledge. It can further enable instructors to provide personalized instruction and exercises for individual students to enhance their learning experiences. We can use psychological theory and statistical models -- Item Response Theory (IRT) and Bayesian Knowledge Tracing (BKT) model has been studied mature in field of traditional intelligent tutoring system (ITS) to model student learning. Our research content of this paper is based on the student’s problem sequences performance, combining IRT and the traditional BKT to improve the prediction accuracy of students’ knowledge state and personalize knowledge tracing model.

Item Response Theory (IRT) [3, 4] predicts a student’s performance on an item based on the difficulty and discrimination of the skill(s) and estimate of the student’s proficiency. Prior work utilized IRT to estimate the static proficiency of knowing a given skill [5], or dynamic changes in overall proficiency [6]. In our paper, we dynamically estimate individual skills required in observed attempts.

Traditional Bayesian knowledge tracing (BKT) [1] estimates the probability that a student knows a skill by observing performance sequences that require it, and applying a model with four parameters for each skill, assumed to be the same for all students: the probabilities knowing skill before practicing it, the probabilities acquiring the skill from one attempt, guess of succeeding item without knowing the skill, and slip of failing despite knowing the skill. Prior work shows that fitting such parameters for individual students can improve the model’s accuracy in predicting student performance [2]. Such per-student parameters, ignore differences between skills; per-skill parameters can also lead to ignore differences between students. Fitting BKT parameters separately instead of each <student, skill> pair risks sparse training data, overfitting, and ignoring overlap of students or skills across pairs.

The rest of this paper is organized as follows: We begin with
the brief review of related work. Section 3 describes the dataset used in our research. In Section 4, we introduce the models used in our paper and detail the IRT-BKT model for assessment of learning outcome. Section 5 presents the prediction results and experimental findings. Finally, we conclude our work in this paper.

Fig. 1. The distribution of skills in C programming course

II. RELATED WORKS

Various research works have studied student outcome modeling in traditional approach. Especially the knowledge tracing model, which was introduced by Corbett and Anderson in 1995, has become the dominant method of modeling student outcomes. Reye [7] showed that the formulas used by Corbett and Anderson in their knowledge tracing work could be derived from a Hidden Markov Model or Dynamic Bayesian Network (DBN). Many other works were introduced to improve the prediction accuracy of the standard knowledge tracing model by using the contextualization of parameters [8] or individualizing the parameters [9].

TABLE I.

<table>
<thead>
<tr>
<th>Data</th>
<th>Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Students</td>
<td>213</td>
</tr>
<tr>
<td>Skills</td>
<td>13</td>
</tr>
<tr>
<td>Items</td>
<td>156</td>
</tr>
<tr>
<td>Submissions</td>
<td>16818</td>
</tr>
</tbody>
</table>

In 2006, Johns and Woolf proposed the Dynamic Mixture Model (DMM) for predicting student knowledge and engagement [13]. They used a Hidden Markov Model [33] like BKT for tracing engagement, but paired it with an Item Response Theory-like model for predicting knowledge. Then Sarah E Schultz [14] proposed the knowledge and affect tracing (KAT) model, which combines two hidden Markov models, BKT and the engagement tracing piece of DMM. This model was able to predict both performance and behavior better than the dynamic mixture model, but did not predict performance as well as standard BKT, perhaps due to over-parameterization.

There were still a deal of Previous studies for modeling learning outcomes [29] in the field of programming languages courses, for example, Michael Yudelson explored [15] automated approaches for extracting domain models for learning programming languages and modeling student knowledge in the process of solving programming exercises. Jussi Kasurinen presented [16] a concept for three-phase measuring method, which are used to observe the student errors, applied programming structures and an application of a Bayesian learning model to determine the programming knowledge. On the other hand, Marc Berges regarded [17] programming ability of a person as latent psychometric constructs and apply the methodology of item response theory (IRT) to assess their manifestations.

Fig. 2. Topologies of Bayesian Knowledge Tracing Model

In our work, we combine item response theory and Bayesian knowledge tracing to model student learning outcomes in studying programming language course, aims to achieve higher estimate accuracy.

III. DATAAETS

Programming language (C language) is the first programming course for the freshmen in the major of computer science and other information related majors. At present, students of our school take programming homework exercises and examinations in an online learning system customized for programming. In our research, we collect information about the freshmen in college who were enrolled in the C language programming course at the fall semester of 2015. As shown in Table 1, the data set consists of student learning outcomes in answering blank-filling or multiple-choice questions.

We classify multiple-choice or blank-filling items into 13 skills with artificial expert statistical knowledge. Because these questions are not complicated, so one item corresponds to only
programming course, which includes 26 items. Type take up least includes 1 item. The whole skill distribution of C language course will be displayed within bar chart as illustrated above.

IV. MODELS

The challenge in predicting student performance is that knowledge state is hidden variable and must be inferred from student performance. Two classes of approaches have been explored, which we’ll refer to as Item Response Theory and Bayesian Knowledge Tracing. In this paper, we propose a Knowledge Tracing Model based on Item Response Theory (IRT-BKT). We begin, however, with a summary of past work.

A. Item Response Theory

The purpose of IRT [30] is to provide a framework for evaluating how well assessments work, and how well individual items on assessments work. The most common application of IRT is in education, where psychometricians use it for developing and designing exams, maintaining banks of items for exams, and equating the difficulties of items for successive versions of exams (for example, to allow comparisons between results over time) [25].

IRT models are often referred to as latent trait models. The term latent is used to emphasize that discrete item responses are taken to be observable manifestations of hypothesized traits, constructs, or attributes, not directly observed, but which must be inferred from the manifest responses. Latent trait models were developed in the field of sociology, but are virtually identical to IRT models.

Item Response Theory (IRT) is a standard framework for modeling student responses dating back to the 1950s [18, 19], were developed by psychometricians to examine test behavior at the problem level (van der Linden & Hambleton 1997). Item Response Theory relates a student’s ability, to her performance on a given problem (Equation 1).

The prerequisite for using the IRT model is that many students have completed a large number of dichotomous items and assigned a priori probability to each student. The one-parameter of IRT is to model the variables between different items, which is named as one-parameter model. Each item has a parameter that represents the difficulty. As such, we extend the parameters of the one-parameter model to a 2-parameter model to get more accurate results.

IRT’s 2-Parameter Logistic model (2PL IRT) estimates the probability \( p(\theta_n | j) \) of student \( n \) already knowing skill \( j \) as a logistic function of student proficiency \( \theta_n \), skill discrimination \( a_j \) and difficulty \( b_j \):

\[
p(\theta_n | j) = \frac{1}{1 + \exp(-a_j(\theta_n - b_j))}
\]

IRT’s 1-Parameter Logistic model (1PL IRT) regard skill discrimination \( a_j \) as a constant \( a \), the model is given in Equation (2):

\[
p(\theta_n | j) = \frac{1}{1 + \exp(-a(\theta_n - b_j))}
\]

Consistent and efficient methods exist for estimating these parameters, in analogy to the estimating parameters of logistic regression model. Methods such as Maximum Likelihood estimation (MLE) or Bayesian approaches can be used to estimate these parameters efficiently.

**Algorithm 1** Markov Chain Monte Carlo Sampling

1) set initial value of parameters: \( \theta^0_n = 0, a^0 = 0, b_j^0 = 0 \)
2) for \( k=1,2,\ldots,T \):
   a) For \( j=1,2,\ldots,N \), sampling \( \theta^k_j \sim N(\theta_j^k, c_0) \)
   Independently
   b) Sampling \( u \sim \text{Uniform}[0, 1] \).
   c) Compute \( R(\theta^k_j, \theta^*_j) = \min \left\{ 1, \frac{p(\theta_j^k)p(Y_j | \theta_j^k, \beta^{k-1})}{p(\theta_j^*p(Y_j | \theta_j^*, \beta^{k-1})}, \frac{\lognormal(\beta^{k-1}, c_0)}{\lognormal(\beta^{k-1}, c_0)} \right\} \)
   if \( u < R(\theta^k_j, \theta^*_j) \) then \( \theta^k_j = \theta^*_j \), else \( \theta^k_j = \theta^*_{j-1} \)
   Next, sampling \( \beta^k \) from \( p(\beta | \theta_j^k, Y) \):
   a) for \( i=1,2,\ldots,M \)
   - Sampling \( \beta_i^1 \sim \lognormal(\beta_i^1, \beta_i^2, c_0^2) \).
   - \( \beta_2 \sim N(\beta_2, \beta_2^1, c_0^1) \), independently
   b) Sampling \( u \sim \text{Uniform}[0, 1] \).
   c) Compute \( R(\beta_i^{k-1}, \beta_i^k) = \min \left\{ 1, \frac{p(\beta_i^k)p(Y_i | \theta_i^k, \beta_i^{k-1})}{p(\beta_i^k)p(Y_i | \theta_i^k, \beta_i^{k-1})}, \frac{\lognormal(\beta_i^{k-1}, c_0^2)}{\lognormal(\beta_i^{k-1}, c_0^2)} \right\} \)
   if \( u < R(\beta_i^{k-1}, \beta_i^k) \), then \( \beta_i^k = \beta_i^k \), then \( \beta_i^k = \beta_i^{k-1} \)
3) Return \( \theta^0, \theta^1, \ldots, \theta^T; \beta^0, \beta^1, \ldots, \beta^T(B > 350) \)

\[
\theta_j = \frac{\sum_{t=B}^{T} \theta_j^t}{T-B}, a_i = \frac{\sum_{t=B}^{T} b_i^t}{T-B}, b_i = \frac{\sum_{t=B}^{T} d_i^t}{T-B}
\]
Estimation, Maximum a Posteriori probability estimation method, Markov Chain Monte Carlo (MCMC) [31] can be used.

A more thorough description of the IRT model, its properties, and the role of each of the parameters can be found in any text on the subject (Baker & Kim 2004; van der Linden & Hambleton 1997). In our work, we adopt MCMC to estimate parameters of IRT, which avoid the sensitivity of initial value, the sampling algorithm is shown in Algorithm 1.

B. Bayesian Knowledge Tracing

Student performances in homework assignment directly reflect their learning outcome. One straightforward way to utilize response data is to let homework correctness rate become the metrics of mastery level. However, the overall correctness rate loses the sequential information of responses. For example, there are two students A and B whose response sequences are \{1, 1, 0, 0\} and \{0, 0, 1, 1\}, whose correctness rates are both 50%, but their learning effects are not the same, considering the sequential information, student B gradually acquires the knowledge likely, while student A may have guessed the first two questions, failed the last two problems unfortunately, therefore the learning outcome of student A should be not as good as student B. Therefore, our work adopts the knowledge tracing model which leverage the response sequence to assess student learning outcomes to tracing the mastery level of student.

Bayesian Knowledge Tracing (KT) comes from the motivation to implement mastery learning [26], where every student is allowed to learn skills at his or her own step and does not continue on to more complex items until mastery of prerequisites has been acquired. It is based on a simplification of the ACT-R theory of skill acquisition [27] and is tasked with inference students’ knowledge level in the Cognitive Tutors, among other ITS. To achieve this end, simpler mastery criterion exist such as N-correct in a row to master, which is used by the ASSISTments Platform [32] in their skill builder problem sets [28]. In a Cognitive Tutor, knowledge, whether declarative or procedural, is defined by fine-grained atomic pieces called knowledge skills in our paper, typically defined by a subject matter expert. Answer steps in the tutor are marked with these skills and a student’s past history of responses indicates his or her level of mastery of the skill. In this context, mastery is inferred to have occurred when there is a high probability (usually \(\geq 0.95\)) that the skill is acquired by the student.

Bayesian Knowledge Tracing (BKT) models the knowledge status of a single skill and is a special case of a Hidden Markov Model (HMM). The traditional knowledge tracing model is
based on 2-state dynamic Bayesian networks, as is shown in figure 2, the knowledge status is changed with the change of the answer sequences in one’s study procedure.

BKT uses two latent states (known and unknown) to model if a student has mastered a particular skill at some point, and two observable states (correct and incorrect) to represent the performance of a particular task. Therefore, the probabilistic model can be described by a set of four probabilities.

There are two performance parameters: guess parameter P(G) represents the fact that the student may sometimes generate a correct response in spite of not knowing the skill. The slip parameter P(S) indicates that even students understand the skill can make an occasional mistake. There are also two learning parameters: the first is initial knowledge P(L0), the probability the student knows the skill before his first answer. The second is the learning rate P(T), the probability a student learn the skill from one-time-step to the next.

The parameters of BKT are learned via Baum-Welch algorithm based on expectation maximization method [20], conjugate gradient search. Then we can use these parameters to update the estimate of the student’s knowledge status through following equation:

\[ p(L_n) = p(L_n|\text{evidence}) + (1 - p(L_{n-1}|\text{evidence})) \ast p(T) \]  

The \( p(L_n) \) is sum of two probabilities: (1) the posterior probability of knowledge was in the learned state conditional on the evidence. (2) probability that student make transition to the learned state if it is not already here. We use a Bayesian inference scheme to estimate the posterior probability \( p(L_{n-1}|\text{evidence}) \) then combine \( p(T) \) we can get \( p(L_n) \) probability the student knows the skill at that time.

The procedure of model learning: (1) Use a large number of student answer sequences, training model parameters. KT model learning based on Bayesian Network which structure known but data is not complete, accordingly, our paper uses Maximization Expectation algorithm to train the model. (2) Leverage the parameters of the trained model to predict students’ mastery level of skills. P(L0) presents the initial mastery level of the skill, then based on the response sequence of students, using the BKT inference algorithm to acquire the mastery level while student finish the last item of the skill. The knowledge status inference algorithm is defined as algorithm 2 as shown above.

C. IRT-BKT

We propose IRT-BKT model [21], whose graphical representation as shown in Figure 3. Obviously, it consists of IRT model and BKT model. Firstly, IRT estimates the probability \( \text{pr}_n \) of student \( n \) already knowing skill \( j \), then fit each skill’s KT parameters \( \text{learn}_n \), \( \text{guess}_j \), \( \text{slip}_j \). The \( Y^{(1)} \) represents students’ response performance at time \( t \) (1 is correct, 0 is incorrect). The latent state \( K^{(t)} \) indicates knowing the skill \( j \) at time \( t \). \( \text{pr}(K^t) = \text{pr}_n \).

Markov Chain Monte Carlo (MCMC) can be used in estimating IRT-BKT’s parameters, we specify their prior distributions as follows:

\[ \theta_n \sim Normal(0, 1) \]  
\[ b_j \sim Normal(0, 1) \]  
\[ a_j \sim Uniform(0, 1) \]  
\[ \text{learn}_j \sim Beta(1, 1) \]  
\[ \text{guess}_j \sim Uniform(0, 0.35) \]  
\[ \text{slip}_j \sim Uniform(0, 0.35) \]

Given observations \( Y \), MCMC estimates vectors \( \theta, a, b, l \) (learn), \( g \) (guess), and \( s \) (slip) with maximum posterior probability:

\[ P(\theta, a, b, l, g, s|Y) \propto P(K^0|\theta, a, b)L(Y|g, s, K) \]  
\[ \times \prod_{t=1}^{T} P(K^t|K^{t-1}, l)p(s)p(l)p(g)p(\theta)p(a)p(b) \]  

IRT-BKT estimates parameters to all data simultaneously, instead of using IRT model to estimate \( \theta, a, b \) within early data and BKT to estimate \( l, g, s \) within later data.

V. EXPERIMENTAL RESULT

In our research, we have 13 skills in the experimental course, with an average of 12 items for each skill. For each skill, there are an average of 213 students complete it. The main goal of this research project was to develop and evaluate a methodology for modeling student learning outcomes of knowledge from blank-filling or multiple-choice questions of programming course. For this purpose, we had to find a more efficient model that would describe the measured outcomes in a suitable way. Additionally, we had to validate the model by evaluating its outcomes. In our research, we apply original model including Item Response Theory and Bayesian Knowledge Tracing to model students learning outcomes, then combine the above two models to IRT-BKT in hope of much higher accuracy rate.

A. Accuracy of Item Response Theory

The whole dataset is split into a training set, a validate set and a test set according to the ratio of 3:1:1. We train model parameters in the training set, use cross validation to determine
the optimal parameters, and use the test set to calculate the metrics. During the model training phase, we used k-fold cross validation, which randomly divides the training dataset into k folds: some are used in constructing a model, and others are used to evaluate the model. Cross validation tests a predictive model without the test dataset, thus helping to gauge whether a model might be over-fitting. We set \( k = 5 \) in our evaluation. During the model evaluation phase, the metrics of recall, precision and AUC (area under the ROC curve), which are broadly used to deal with imbalance dataset. The experimental result is presented in table 2. The highest AUC achieves 0.783 with accuracy of 0.763 we can conclude.

\[
\begin{array}{|c|c|c|}
\hline
\text{Metrics} & 1PL & 2PL \\
\hline
\text{Accuracy} & 0.785 & 0.763 \\
\text{Recall} & 0.832 & 0.793 \\
\text{Precision} & 0.513 & 0.598 \\
\text{AUC} & 0.769 & 0.783 \\
\hline
\end{array}
\]

Prediction accuracy of 1PL and 2PL are shown in Figure 4. Obviously, 2PL model outperforms 1PL model at metric of prediction accuracy at least slightly for all 13 skills. So we can adopt the students’ proficiency from 2PL model as students’ mastery level.

In Figure 5 the item characteristic curves for all items belongs to Function that are included in the model are shown.

According to the definition of the 2PL model, they only differ in their level of difficulty and discrimination. This is expressed in the figure by a shift on the x-axis, which shows the latent parameter on a scale of -4 to 4. All curves are parallel and only differ in the value of the latent parameter at the probability of 50% for rating a code item with yes. The probability that an individual with a specific value of the latent parameter has solved a specific item is drawn on the y-axis. As shown in Figure 5, the simplest item is item3 which represents the use of function calls about arithmetic operators. The underlying concept is simple to understand. The next concept in the ranking is item4 which indicates the use of definition of function calls. Next, the items item2 and item1 indicate the use of function calls about increment operator. Regarding the last item item5, the most difficult item according to the 2PL model is the use of statement of function calls which lead to errors easily. The rest of items belong to other skills can also be acquired for the same reason.

### B. Accuracy of Bayesian Knowledge Tracing

The experimental dataset consists of student responses to homework problems of each skill. We had 13 skills in total course. Only the items within the homework with the exactly same skill tag were used. Each item had an average of 213 students and each student completed an average of 156 items.

We used the students’ responses to the last item of each skill as test data, while the other responses as training data. For each skill, we train Bayesian knowledge tracing models, we first constructed the appropriate sized dynamic Bayesian network, where the size of student nodes corresponded to the number of students completing items. Then the parameters were learned using Expectation Maximization (EM) algorithm. Note that the initial values for learning rate, guess and slip parameters of both models were set randomly, BKT model set the single parameter randomly and allowed it to be adjusted by EM.

After the parameters were learned, we estimated the performance of every student by entering the responses as evidence to the Bayesian network, except the last one. Then the probability of the student answering the last question correctly was computed and converted to binary one by a certain threshold so that it could be compared to the actual response later.

Figure 6 exhibited the change of student’s mastery level along with tracing the performance of items. Green node represent current performance is correct, red node is incorrect, which indicated the mastery level increased when green nodes occurred, decreased met red nodes. Prediction for the probability of answer last item correct is mapped to a colored vertical axis.

![Fig. 7. Prediction accuracy of three different models](image)
C. Validity of IRT-BKT

The former n-1 questions of each skill are used as training data to train the traditional Bayesian knowledge tracking model and the IRT-BKT model respectively. Using the last question as the test data to test the accuracy of the model, and then compared with the IRT model and BKT model, the results shown in Figure 7. We obtained IRT-BKT model outperforms Item response theory and Bayesian knowledge tracing, the validity of the new model was verified. Learning outcomes of Students who studying programming course computed by IRT-BKT model including students’ mastery level for specific skill is our ultimate task. In summary, the average accuracy of the IRT–BKT model reaches as high as 82%, which demonstrates that the proposed new model is of high effectiveness in learning outcome assessment.

VI. CONCLUSION

In our work, we research in the field to model student learning outcomes in studying programming language course. We proposed a new IRT-BKT model that uses IRT to estimate students’ initial knowledge of a skill based on its difficulty and discrimination and their overall proficiency, and KT to model learning over time. It outperforms BKT and IRT by combining information about both. IRT-BKT estimates every probability Knew(student, skill) without requiring training data for every <student, skill> pair, because it can estimate student proficiency based on other skills, and skill difficulty and discrimination based on other students.

There remain a number of issues to be investigated in future work. Future work should compare IRT-BKT to other methods. We should extending IRT-BKT to trace multiple subskills use considerably fewer parameters than prior methods [22, 23], thanks to combining IRT and BKT, we can try other manner for better effects.

ACKNOWLEDGMENT

This work was supported in part by grant from State Key Laboratory of Software Development Environment (Funding No. SKLSDE-2015ZX-03) and NSFC (Grant No. 61532004).

REFERENCES

