Advanced Models for Student Knowledge Estimation

Mouhssine Rifaki
mouhssine.rifaki@psl.eu

1 Introduction

This paper presents my effort to model students' evolving knowledge states during skill acquisition using Bayesian Knowledge Tracing (BKT). Specifically, the BKT Hidden Markov Model (HMM) is employed to predict the probability of correctly applying a skill as a function of the number of prior opportunities to practice that skill. This modeling approach utilizes synthetic data generated based on BKT and Item Response Theory (IRT) assumptions, which simulate a student's knowledge trajectory and provide a contextual estimation of these probabilities. The process begins with a predefined prior probability representing whether the student initially knows a given concept. The models are also evaluated on two real-world datasets, discussed later in the paper.

Five models were trained to estimate student knowledge from observed responses, in addition to a baseline approach. These models include Bernoulli, Logistic Regression, Average Response, Bayesian Knowledge Tracing (BKT), and Clustered BKT. The results indicate that the BKT and Clustered BKT models consistently outperformed the other approaches across all four datasets.

2 Related Work

Since its introduction, the Bayesian Knowledge Tracing (BKT) model has been widely used in studies of student learning and in various intelligent tutoring systems. Its flexibility allows for more complex configurations to accommodate diverse learning models. Effective knowledge tracing can enable personalized learning experiences by recommending resources tailored to individual student needs and skipping content that may be too easy or too difficult. This approach facilitates efficient adaptive learning.

Researchers have primarily utilized BKT in two forms: the Hidden Markov Model (HMM) and the Knowledge Tracing Algorithm [1]. The HMM version of BKT predicts the probability that a student will correctly apply a skill when given the opportunity [2]. The simplicity of the BKT model allows for analytical solutions to the HMM.

Further advancements in knowledge tracing include Deep Knowledge Tracing (DKT), proposed by Piech et al., which demonstrates significantly better results than traditional models [3]. DKT employs flexible recurrent neural networks (RNNs) to model knowledge tracing over time. These models represent latent knowledge states and their temporal

dynamics using large vectors of artificial neurons. Unlike traditional BKT, which relies on hard-coded initial values, DKT learns latent variable representations of student knowledge directly from the data.

3 Datasets

I tested the models on four different datasets, two of which were synthetic. Below is the description of each dataset.

- **Synthetic data generated according to IRT assumptions:** Virtual students learning virtual concepts were simulated, and the accuracy of predictions for their responses was tested in this controlled setting. A total of 500 synthetic students and 10 concepts were generated. Each question was associated with a specific concept and difficulty level. Using the IRT (Item Response Theory) model's "squeezed sigmoid" function, the probability of a student correctly answering a question was modeled as follows:

$$P(\text{correct}|\text{skill}, \text{difficulty}) = 0.25 + 0.75 \cdot \frac{1}{1 + e^{\text{difficulty}-\text{skill}}}.$$

  For a fixed concept difficulty level, this probability could take on only one of two values, corresponding to knowledge states of 1 (know) and 0 (don't know). Students had closely correlated prior per-concept knowledge and learning rate parameters.

- **Synthetic data generated according to BKT assumptions:** Based on BKT assumptions, which assert that students do not forget a concept once learned, students' knowledge of different concepts was modeled using a Markov chain between "don't know" and "know" states. The emission states for the HMM were "correct" or "incorrect" (i.e., 1 or 0). Sequences of emissions were generated from the Markov chain for 500 virtual students and 10 virtual concepts.

- **KDD Bridge to Algebra 2006–2007:** This real dataset [4] was sourced from the KDD Cup, an annual Data Mining and Knowledge Discovery competition organized by ACM SIGKDD. The full dataset contained 808 concepts and 5968 students. To reduce execution time, a subset of 10 concepts and 500 students was used. Two thresholds, t_1 and t_2, were applied to ensure sufficient data for each concept. The first threshold, t_1, checked the number of students who answered questions for a particular concept, and concepts with fewer than t_1 students were dropped. The second threshold, t_2, ensured each student answered a sufficient number of questions per concept; students who answered fewer than t_2 questions for a concept were dropped. The models were then tested on the remaining data, ensuring a balanced subset.

- **Assistments:** Similar to the KDD Cup data, a subset of this real dataset [5] was used. The full dataset included 125 concepts and 4218 students. A subset of 10 concepts and 1678 students was selected, applying the same thresholds for students per concept (t_1) and questions answered per student per concept (t_2) to ensure

sufficient data for testing.

The data structure used in this study is organized as follows: A group of n students answers a set of p problems from c different concepts. Each student's responses to problems for each concept are stored as a vector $\mathbf{a}$, where $\mathbf{a} \in \mathbb{R}^p$ and $a_i \in \{0, 1\}$, with 0 representing an incorrect answer and 1 representing a correct answer. For example, if a student answers a series of 6 questions for the concept "long division" and gets the first 3 incorrect and the last 3 correct, their response vector would be $[0, 0, 0, 1, 1, 1]$. Each concept is represented as a matrix $\mathbf{M} \in \mathbb{R}^{n \times p}$, where each row corresponds to a student and each column corresponds to a question.

4 Methods

The following problem is addressed: given student responses to a series of questions on a number of concepts, which concepts does the student know? Framed this way, student knowledge on a particular concept can be modeled as a latent variable, while answers to questions on that particular concept can be modeled as observed variables. To address this, six models were trained to estimate student knowledge from observed responses. Each model differs in its underlying assumptions. The major variations are as follows:

- The Baseline, Bernoulli, Logistic Regression, and Bayesian Knowledge Tracing (BKT) models assume that students are identical (i.e., they do not model per-student parameters). In contrast, the Average Response and Clustered BKT models account for differences between students.

- The Baseline and Average Response models do not consider concepts (i.e., they do not model concepts explicitly). The other four models train separate models for each concept.

Each model provides an estimate of a student's knowledge state on a particular concept at two key points: (1) before the student has attempted any questions on that concept and (2) immediately after the student answers a question on that concept. These estimates are used to calculate the probability that the student will answer the question correctly, denoted as $P(\text{correct})$. This probability is then used to predict the student's response.

Finally, the predicted answer sequence is compared to the actual answer sequence to evaluate the effectiveness of each model.

4.1 Baseline

A simple baseline model was implemented where $P(\text{correct}) = 1$. This model predicts that all students answer every question correctly. It served as a lower bound for comparison against all other models.

4.2 Bernoulli

A separate Bernoulli model was trained for each concept in the data. In this model, $P(\text{correct}_k) = \phi_k$, where ϕ_k is the percentage of correct answers on that concept in the training data. If there are c concepts in the training data, c distinct ϕ's are learned. This model assumes that all students are identical; hence, predictions for a question on concept k use ϕ_k regardless of the student.

4.3 Logistic Regression

For each concept, $P(\text{correct})$ is modeled using a logit link with the number of questions the student has seen for that concept as the sole independent variable. This model also assumes that students are identical.

4.4 Average Response

For each concept k, $P(\text{correct}_k)$ is modeled per student as the student's percentage of correct answers on that concept up to the present point. For instance, if a student is presented with a question on the concept "Quadratic Equation," the probability is calculated as:

$$P(\text{correct}_{\text{Quadratic Equation}}) = \frac{\text{Number of prior correct answers on 'Quadratic Equation'}}{\text{Number of total prior questions on 'Quadratic Equation'}}.$$

4.5 Bayesian Knowledge Tracing

In Bayesian Knowledge Tracing (BKT), one model is fit per concept, treating all students as identical. Furthermore, a two-state learning model is assumed. Specifically, for a given concept, it is assumed that the student is either in a "knowing" state or a "not-knowing" state. A student can transition from not knowing to knowing each time they answer a question on that concept. In BKT, it is assumed that once a student learns a concept, they never forget it. Additionally, the model accounts for the probability that a student answers incorrectly despite knowing the concept (called a slip) and the probability that a student answers correctly despite not knowing the concept (called a guess). The parameters for this model are summarized as follows:

- $P(L_c^k)$: The probability that the student knows concept c when answering the kth question on that concept.
- $P(L_c^0)$: The probability that the student knows concept c before answering any questions on that concept (a special case of the previous probability).
- $P(T^c)$: The probability of learning concept c when answering a question on that concept.

- $P(G^c)$: The probability of guessing correctly on concept c if in the not-knowing state.
- $P(S^c)$: The probability of answering incorrectly (slipping) on concept c despite being in the knowing state.

There are two distinct stages to using the BKT model. First, the parameters for each model are learned from the training data. Second, these models are used to infer a student's knowledge state as they work through questions.

4.5.1 BKT as a Hidden Markov Model

The model described in the previous sections is a Hidden Markov Model (HMM) with the following parameters. Modeling the problem as an HMM allows the use of the EM algorithm to learn prior probabilities, transition probabilities, and emission probabilities from the training data.

(a) Priors (Π)

known	$p(L_0)$
unknown	$1 - p(L_0)$

(b) Transitions (A) to known & to unknown

from known	1	0
from unknown	$p(T)$	$1 - p(T)$

(c) Observations (B) right & wrong

known	$1 - p(S)$	$p(S)$
unknown	$p(G)$	$1 - p(G)$

Table 1: BKT parameters in matrix form.

4.5.2 Updating Student Knowledge

Given an observation of the student's response at time opportunity k (correct or incorrect) on concept c, the probability $P(L_c^k)$ that a student knows concept c is calculated using Bayes' rule. When a correct response is observed, this probability is given by:

$$P(L_c^k|\text{correct}_c^k) = \frac{P(L_c^k) \cdot (1 - P(S^c))}{P(\text{correct}_c^k)}.$$

When an incorrect response is observed, this probability is given by:

$$P(L_c^k|\text{incorrect}_c^k) = \frac{P(L_c^k) \cdot P(S^c)}{P(\text{incorrect}_c^k)}.$$

Finally, the student's knowledge of concept c is updated based on their interaction with the system. This updated estimate is the sum of two probabilities: the posterior probability that the student already knew the concept (based on the evidence) and the probability that the student did not know the concept but was able to learn it:

$$P(L_c^k) = P(L_c^{k-1}|\text{evidence}_c^{k-1}) + (1 - P(L_c^{k-1}|\text{evidence}_c^{k-1})) \cdot P(T^c).$$

4.5.3 Making Predictions with BKT

The probability of a student answering a question on concept c correctly at time opportunity k is given by:

$$P(\text{correct}_c^k) = P(L_c^k) \cdot (1 - P(S^c)) + (1 - P(L_c^k)) \cdot P(G^c).$$

5 Clustered Bayesian Knowledge Tracing

In Clustered BKT, students are first clustered into distinct groups, and a distinct set of HMMs is trained for each group. For instance, if students are clustered into g groups and there are c concepts in the data, a total of $g \cdot c$ models are trained (as opposed to the c models in standard BKT).

Initially, students were represented as n-dimensional vectors, and k-means clustering was used to divide them into groups. However, when this approach did not provide any additional gains over BKT, a pseudo-clustering algorithm was implemented as follows. Students in the training data were first divided using a median split based on the percentage of total correct answers across all concepts. Subsequently, one HMM per group per concept was trained, resulting in double the number of models compared to standard BKT.

This method assumes the presence of high-achieving and low-achieving students in the data. It further assumes that the BKT parameters that best model the high-achieving group differ from those that best model the low-achieving group.

5.1 Making Predictions with Clustered BKT

Predictions for novel answer sequences were made such that at each index k, the percentage of correct answers up to k determined which HMM's predicted state sequence was used to predict the next emission. Once the model was selected, the method described in the BKT section above was used to compute $P(\text{correct}_k)$.

6 Results

Each of the five models was evaluated on all four datasets, resulting in a total of 20 combinations. Monte Carlo cross-validation was used for evaluation, where at each iteration, the data was pointwise randomly split into a $90 : 10$ training-to-validation ratio. In each dataset, each training point represented a student sequence of correct or incorrect answers. At each iteration, a random set of students (real or synthetic, depending on the dataset) was used to train the algorithm, while a separate random set of students was used for evaluation.

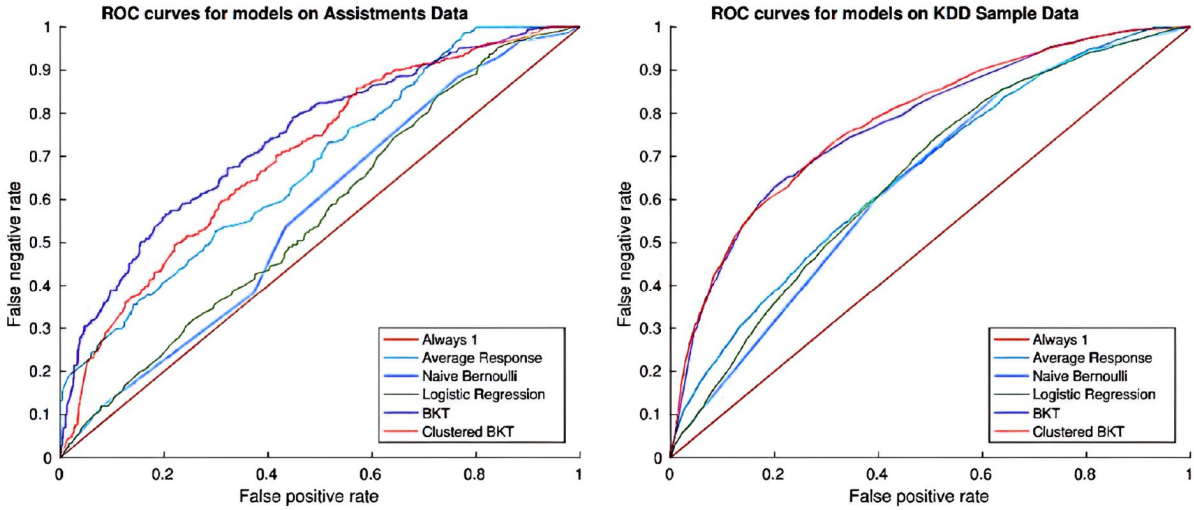

Figure 1: Comparison of ROC curves for the Assistments and KDD datasets. This figure demonstrates performance insights across the two datasets.

Each model was trained on the training set and evaluated on the validation set as follows: each sequence was fed into the model answer-by-answer, and at each answer index i, the model output its estimate of the probability that the $i+1$-th answer would be correct (i.e., a 1).

The model's outputs formed a vector of probabilities p_i, where $p_i \in [0,1] \forall i$, which was compared to the true sequence of outputs x_i, where $x_i \in \{0,1\} \forall i$. Metrics for each model were calculated relative to these vectors and averaged over all student sequences in the validation set. The mean-squared error (MSE) was calculated as:

$$\text{MSE} = \sum_{i=1}^{n} (x_i - p_i)^2.$$

For the error rate, a threshold of 0.5 was set, and the proportion of answer indices where the rounded probability did not match the true output was reported. For the AUROC curve, Matlab's library was used to generate false-positive and true-positive rates for various threshold values, and the area under the resulting curve was calculated.

Results shown in the graphs (Figure 1) represent 100-iteration Monte Carlo cross-validation, with error bars indicating the sample standard deviation of the reported statistics. Meta Hyperparameter optimization was unnecessary, as the models used were parameterized solely based on empirical fitting. For HMM-based BKT models, the EM algorithm was used and restarted 10 times per iteration to ensure convergence to a global optimum.

The simplest useful models, Naive Bernoulli and Logistic Regression, performed consistently well on the synthetic datasets (0.15 and 0.1 MSE on synthetic BKT and synthetic IRT, respectively) and on the KDD dataset (0.15 MSE). However, their performance was significantly worse on the Assistments dataset (0.25 MSE). Notably, Logistic Regression outperformed Naive Bernoulli on synthetic data but not on real-world data, suggesting that the assumption of time-dependency—where students become more likely to answer related questions correctly over time—did not hold.

BKT models performed best on all datasets (0.06, 0.1, 0.13, and 0.2 MSE on IRT, BKT, KDD, and Assistments, respectively), validating their use despite their relative complexity. Interestingly, the Clustered BKT model performed almost identically to the unclustered model, indicating that clustering by student response patterns did not provide additional useful information. As expected, the Always-1 and Average Response models performed worse than the other models.

Other quality metrics (AUROC and average error) confirmed the overall superiority of BKT models, though the advantage for KDD and synthetic IRT data was relatively small (< 0.05 better than the next best models). Across all datasets, BKT models exhibited higher ROC curves than the other models.

There was no evidence of significant overfitting, as the standard deviations across Monte Carlo cross-validation trials were small relative to the means. This consistency suggests that the models generalized well to unseen data and did not overfit to the training set. However, the large performance differences between the two real-world datasets (Assistments and KDD) indicate that model performance on other real-world datasets could vary significantly.

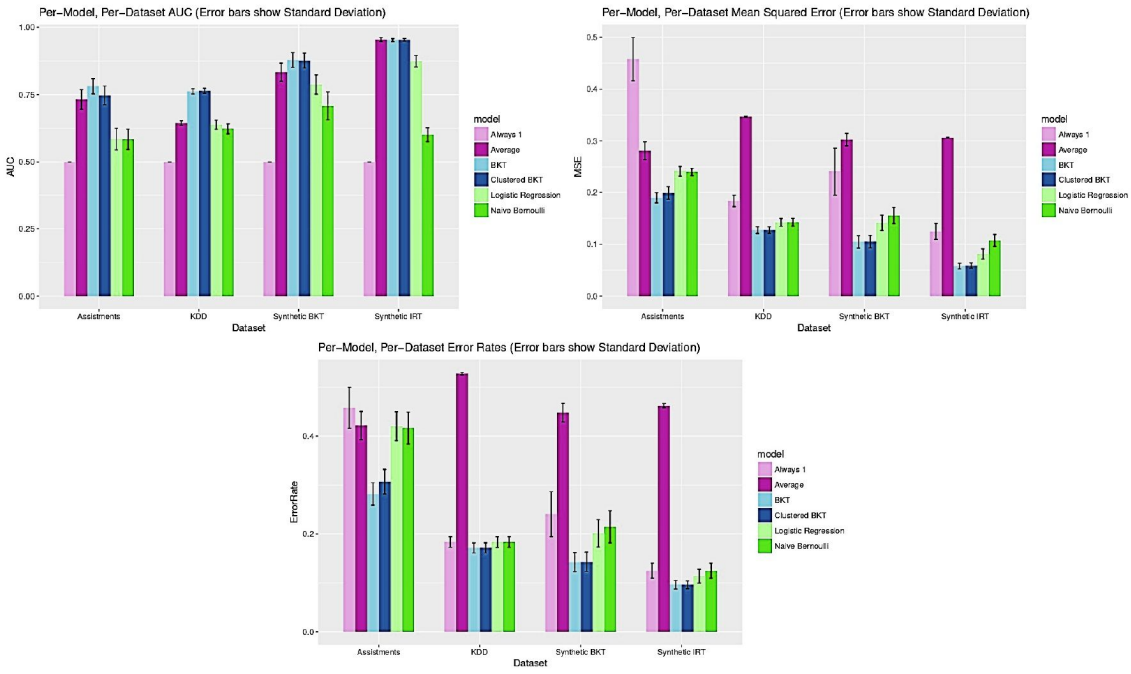

Figure 2: Per-model metrics: AUROC, MSE, overall error rate.

7 Conclusion

Only BKT and Clustered BKT consistently outperformed the baseline across all datasets. The Average Response model performed nearly as well as BKT and Clustered BKT on the Assistments and Synthetic datasets but performed significantly worse than BKT on the KDD dataset. Although the results indicate that BKT is the most reliable model for predicting student knowledge across multiple datasets, its improvement over the baseline was marginal in some cases.

Clustered BKT and BKT achieved nearly identical performance across all datasets. This outcome was unexpected, suggesting that in the future, a different set of features should be selected to improve the effectiveness of clustering.

References

[1] B. van De Sande, "Properties of the bayesian knowledge tracing model," *Educational Data Mining Journal*, pp. 1–10, 2023.

[2] J. Beck and K.-m. Chang, "Identifiability: A fundamental problem of student modeling," vol. 4511, 07 2007, pp. 137–146.

[3] C. Piech, J. Spencer, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein, "Deep knowledge tracing," 06 2015.

[4] J. Stamper, N.-M. A. R. S. G. G., and K. K., "Bridge to algebra 2008-2009: Challenge data set from kdd cup 2010 educational data mining challenge," 2008, challenge data set from KDD Cup 2010.

[5] N. Heffernan and C. Heffernan, "The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching," *International Journal of Artificial Intelligence in Education*, vol. 24, 12 2014.