KVFKT: A New Horizon in Knowledge Tracing with Attention-Based Embedding and Forgetting Curve Integration

Quanlong Guan[1,6], Xiuliang Duan[1]*, Kaiquan Bian[1], Guanliang Chen[4], Jianbo Huang[3]†, Zhiguo Gong[2,6], Liangda Fang[1,5]

[1]Jinan University, Guangzhou, China [2]University of Macau, Macao, China
[3]South China University of Technology, Guangzhou, China
[4]Monash University, Melbourne, Australia [5]Pazhou Laboratory, Guangzhou, China
[6]Guangdong-Macao Advanced Intelligent Computing Joint Laboratory, Zhuhai, China

*†**Correspondence:** dxl1001@stu2022.jnu.edu.cn, jbhuang@scut.edu.cn

Abstract

The knowledge tracing (KT) model based on deep learning has been proven to be superior to the traditional knowledge tracing model, eliminating the need for artificial engineering features. However, there are still problems, such as insufficient interpretability of the learning and answering processes. To address these issues, we propose a new approach in knowledge tracing with attention-based embedding and forgetting curve integration, namely KVFKT. Firstly, the embedding representation module is responsible for embedding the questions and computing the attention vector of knowledge concepts (KCs) when students answer questions and when answer time stamps are collected. Secondly, the forgetting quantification module performs the pre-prediction update of the student's knowledge state matrix. This quantification involves calculating the interval time and associated forgetting rate of relevant KCs, following the forgetting curve. Thirdly, the answer prediction module generates responses based on students' knowledge status, guess coefficient, and question difficulty. Finally, the knowledge status update module further refines the students' knowledge status according to their answers to the questions and the characteristics of those questions. In the experiment, four real-world datasets are used to test the model. Experimental results show that KVFKT better traces students' knowledge state and outperforms state-of-the-art models.

1 Introduction

KT (Corbett and Anderson, 1994; Lee et al., 2023; Rasch, 1993) is the task of modeling a student's learning state and predicting their future performance based on their question-solving behavior over time (Duan et al., 2024). In recent years, the scientific community has paid significant attention to KT, and it has been integrated into people's daily learning activities (Abdelrahman et al., 2023; Ni et al., 2023). Furthermore, a current trend in the development of deep learning KT models is to incorporate forgetting features, allowing students' forgetting behavior to be considered in knowledge tracing (Abdelrahman and Wang, 2022).

Although conventional KT methods have achieved empirical success (Liu et al., 2019; Pardos and Heffernan, 2010; Yudelson et al., 2013; Corbett and Anderson, 2005), they often overlook the influence of process-driven factors within the student's learning process. First, learning is a dynamic process, and each student's learning gains are often different and implicit. It is critical to precisely measure the knowledge gained after each question response (Khajah et al., 2014; Li et al., 2022). Second, while responding, students may speculate. The amount of time students take to respond to questions is directly related to the guessing coefficient (Bai et al., 2024). Third, forgetting must be considered in KT when students neglect previously learned knowledge over an extended period (Zhou et al., 2024). As shown in Figure 1, even if students have strong knowledge of the "isosceles triangle" KC, they may still give the wrong answer if q_6 takes too long to answer. Therefore, we argue that when students take a long time to answer questions, their final results are likely to be based on guessing (Zhao et al., 2023). As the guessing coefficient increases, students' answers become more erratic. Moreover, forgetting should be considered when students study prior KCs, as this directly affects their proficiency in the related KCs (Han et al., 2023; Cui et al., 2024). Thus, we believe that there is a correlation between the state information of a student answering a question—such as response time, guessing coefficient, and memory forgetting.

To address these issues, we propose a new approach in knowledge tracing with attention-based embedding and forgetting curve integration, namely KVFKT. First, the embedding representation module is responsible for embedding the ques-

Proceedings of the 31st International Conference on Computational Linguistics, pages 4399–4409
January 19–24, 2025. ©2025 Association for Computational Linguistics

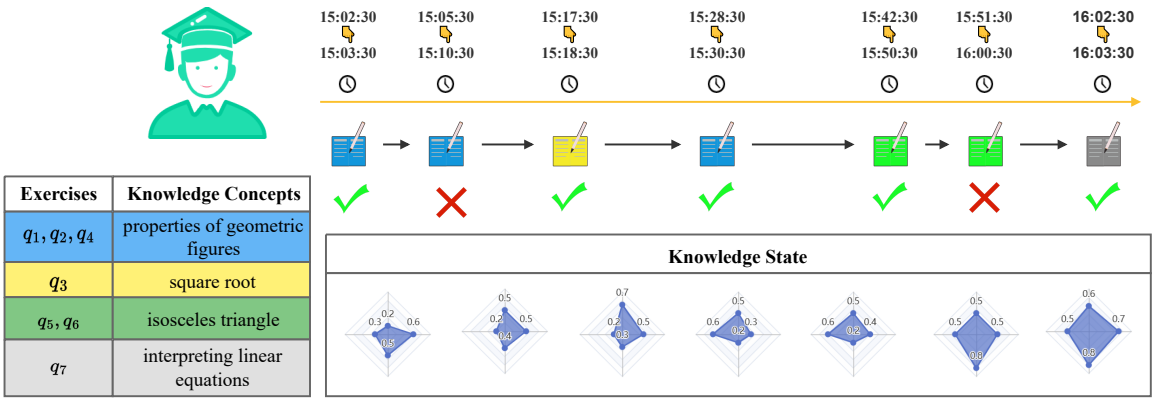

Figure 1: A case of a student's question sequence, where he has already done seven questions and is going to answer the next question. During the learning process, different kinds of side information are recorded in addition to the answer in the system.

tions and computing the attention vector of KCs as students' responses and time stamps are collected. Second, the forgetting quantification module updates the students' knowledge state matrix by calculating the time intervals and corresponding forgetting rates of relevant KCs, based on the forgetting curve. Third, the answer prediction module generates responses by considering the students' knowledge state, the guessing coefficient, and the difficulty of the questions. Finally, the knowledge state update module further refines the students' knowledge state based on their responses and the specific characteristics of the questions answered. In the experiment, four real-world datasets are used to test the model. The experimental results show that KVFKT better traces students' knowledge states and outperforms state-of-the-art models.

2 Related Works

2.1 Knowledge tracing

The most representative methods for knowledge tracing are deep learning-based methods (Piech et al., 2015; Liu et al., 2024; Ma et al., 2024). These deep learning-based methods can be categorized into three main groups: single-state methods, multi-state methods, and attention-based methods. Single-state methods maintain one vector to represent students' knowledge states. Deep Knowledge Tracing (DKT) (Piech et al., 2015) is a typical example of a single-state method. Multi-state methods, on the other hand, maintain multiple vectors to represent a student's knowledge state. Attention-based methods use attention mechanisms to identify the correlation between the question to be predicted and historical questions (Pandey and Srivastava, 2020), predicting based on a student's past performance. A typical example is Attentive Knowledge Tracing (AKT) (Ghosh et al., 2020), which learns the context representation of the questions and answers using two attention encoders and then utilizes the attention mechanism to recall prior learning relevant to the current activity. Self-Attentive Knowledge Tracing (SAKT) (Pandey and Karypis, 2019) employs self-attention (Vaswani et al., 2017) to weigh historical performances. Separated Self-Attentive Neural Knowledge Tracing (SAINT) (Choi et al., 2020) is a typical transformer-based structure that embeds the questions in an encoder and predicts the responses in a decoder. Although these methods have provided elegant solutions by distributing the vector representations of a KC, they still lack interpretability. In contrast, some traditional models, such as Item Response Theory (IRT) (Rasch, 1993) and the Three-Parameter Logistic (3PL) model (Lo, 2008), have parameters that offer direct psychological interpretations. These interpretable parameters are also reflected in the KVFKT model.

2.2 Forgetting curve

In the learning process, forgetting is a widely recognized phenomenon (Markovitch and Scott, 1988; Choffin et al., 2019). According to the hypothesis of the forgetting curve, students' memory naturally declines during the learning process, often leading to decreased proficiency in KCs (Averell and Heathcote, 2011; Guan et al., 2025). Trace decay theory proposes that forgetting occurs due to the gradual disappearance of memory traces (Ricker et al., 2016), with deeper original traces leading to a slower rate of forgetting. Memory is stored in im-

print cells, and forgetting happens when these cells cannot be reactivated (Ryan and Frankland, 2022). Ebbinghaus (Ebbinghaus, 1885) used a non-linear function to connect the observed memory retention probability with the interval between learning and testing time to explain the forgetting curve. The forgetting curve can be modeled as a power law function, where memory strength initially drops rapidly and then decays more slowly over a longer period. Therefore, how to effectively integrate educational psychology theory and neurology theory to more comprehensively model learning and forgetting behaviors remains a pressing challenge. DKT-Forgetting attempts to improve the DKT model by considering factors such as the number of times students repeat learning, the time interval since their last review of the same KC, and the time intervals from prior learning (Nagatani et al., 2019). The Exercise-correlated Knowledge Proficiency Tracing (EKPT) model takes into account the correlation between exercises involving the same knowledge concepts (Huang et al., 2020). The Knowledge Tracing Model With Learning and Forgetting Behavior (LFBKT) separates the learning process into two stages: knowledge acquisition and knowledge retention. The knowledge retention module includes both knowledge absorption and knowledge forgetting (Chen et al., 2022; He et al., 2025). Finally, in knowledge tracing, considering the one-to-many relationship between KCs and questions, we attribute the forgetting phenomenon to specific KCs rather than questions, which enables more effective updates to students' knowledge states.

3 Problem Definition

3.1 Question answering sequence

$x_t = (\{q_1, c_1, time_1, at_1, y_1\}, ..., \{q_i, c_i, time_i, at_i, y_i\}, ..., \{q_{t-1}, c_{t-1}, time_{t-1}, at_{t-1}, y_{t-1}\})$ represents a student's question-answering sequence before time t (excluding time t). In this sequence, $q_i \in Q$ denotes the i-th element of the set Q, and Q denotes a set of questions. The $c_i \in \mathbb{R}^{d_c}$ denotes the set of concepts related to q_i. The $time_i$ is the current time at step i. The at_i represents the answer time, calculated as $time_{i+1} - time_i$. The $y_i \in \{0, 1\}$, with 0 indicating an incorrect answer and 1 indicating a correct answer to question q_i. A question-answering sequence contains the history of a student's interactions. We formulate the KT problem as a sequence learning problem(Zhang et al., 2017).

3.2 Knowledge tracing

Given a question answering sequence x_t and an input question $q_t \in Q$, the knowledge tracing problem is to predict the probability that q_t is answered correctly. As a student answers questions, the knowledge state undergoes evolution. Each student is thus associated with a sequence of knowledge states $(s_1, ..., s_t)$ where $s_1, ...s_t$ refer to the knowledge states at the time steps $1, ..., t$, respectively. In this work, we tackle the KT problem by developing a machine learning model $\mathcal{M}_\theta$, parameterized by θ. Concretely, given a question answering sequence x_t and an input question $q_t \in Q$, a KT model $\mathcal{M}_\theta$ can trace the knowledge state s_t at each time step t and generate as output the probability p_t of correctly answering the question q_t, i.e. $p_t = p(y_t = 1 | x_t, s_t, q_t)$.

4 The KVFKT Model

In this section, we will introduce the KVFKT model in detail. As shown in Figure 2, each learning phase of KVFKT consists of four modules. Firstly, the embedding representation module is responsible for embedding the questions and computing the attention vector of Knowledge Concepts (KCs) as students answer questions and answer timestamps are collected. Secondly, the forgetting quantification updates the students' knowledge state matrix before prediction. It quantifies the forgetting process by calculating the time intervals and associated forgetting rates of relevant KCs, based on the forgetting curve. Thirdly, the answer prediction module generates responses by considering the students' knowledge state, the guessing coefficient, and the difficulty of the questions. Finally, the Knowledge status update module further refines the students' knowledge state based on their responses to the questions and the specific characteristics of the questions answered.

4.1 Embedding representation

To obtain more accurate learning gain information, we incorporate a practice embedding matrix to combine the practice and concept features into a cohesive vector representation. Given input question $q_t \in \mathbb{R}^{d_q}$ and KC $c_t \in \mathbb{R}^{d_c}$, we get the question embedding vector $Q_t \in \mathbb{R}^{d_k}$ and KC embedding vector $C_t \in \mathbb{R}^{d_m}$ through embedding matrices $Q \in \mathbb{R}^{d_q \times d_k}$ and $C \in \mathbb{R}^{d_c \times d_m}$, where d_q is the number of questions set, d_c is the number of concepts set and d_k is a hyperparameter. After obtain-

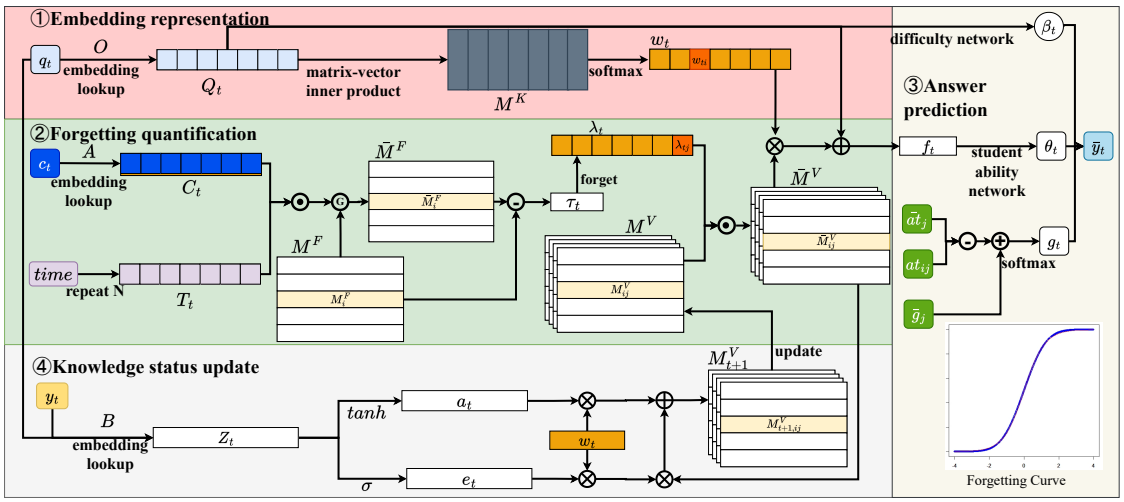

Figure 2: KVFKT Model Structure, where M^K represents the mapping matrix that encodes the relationship between concepts and questions, capturing the interaction between memory and attention. This matrix remains constant throughout the entire process; M^F denotes the forgetting matrix, which tracks the timestamp of each student's most recent review of the KCs, helping to account for forgetting behavior; M^V refers to the knowledge state matrix, representing the student's current knowledge state at each time step.

ing the question embedding vector Q_t, we query the key memory matrix $M^K \in \mathbb{R}^{d_m \times d_k}$ in the KVFKT model. The query result is the weighting of how much attention should be paid to each value memory slot. This attention weight $w_t \in \mathbb{R}^N$ is computed by the softmax activation of the inner product between Q_t and each key memory slot M_i^K:

$$w_{t,i} = Softmax(M_i^K Q_t). \tag{1}$$

4.2 Forgetting quantification

To simulate students' knowledge degradation over time, we use the forgetting curve in KT. For the answer interval, its knowledge retention rate is $(\lambda = e^{-\frac{\tau}{S}})$, where S is a user-defined parameter, which is a complete forgetting cycle. For interval, its calculation is:

$$\Omega_t = C_t \cdot T, \tag{2}$$

$$\bar{M}_i^F = G(\Omega_t, M_i^F), \tag{3}$$

$$G(x_i, y_i) = \begin{cases} y_i & , \quad x_i < 0.5 \times T_i \\ x_i & , \quad other \end{cases}, \tag{4}$$

$$\tau_j = \bar{M}_{i,j}^F - M_{i,j}^F (j \in [1, d_m]), \tag{5}$$

where $T \in \mathbb{R}^{d_m}$ is current time vector and T_i is current timestamp. Next, we will point multiply it with the KC embedding vector C_t to get the intermediate vector $\Omega_t \in \mathbb{R}^{d_m}$. Then, we perform $G(.)$ function processing on the vector M_i^F in row i of forgetting matrix $M^F \in \mathbb{R}^{d_s \times d_m}$ and Ω_t to obtain the updated forgetting matrix $\bar{M}_i^F$. For each element in the matrix, when it is less than $0.5 \times T_i$, we default that the KCs represented by the vector slot are not reflected in the question. In such cases, we will not update the corresponding elements of forgetting matrix, conversely, we will update them to the current timestamp. Finally, we get the τ_j vector by subtracting $\bar{M}_i^F$ vector and M_i^F vector elements one by one. Based on this, we update the value matrix:

$$\lambda_j = e^{-\frac{\tau_j}{S}}, \tag{6}$$

$$\bar{M}_{i,:,j}^V = Softmax(M_{i,:,j}^V \cdot \lambda), \tag{7}$$

where λ_j is the forgetting rate of the corresponding knowledge slot calculated by the model. We perform element-wise multiplication of λ with the original value matrix M^V, where the i-th student's knowledge state matrix is denoted as $M_i^V \in \mathbb{R}^{d_m \times d_v}$. Afterward, we apply the softmax function to obtain the updated value matrix $\bar{M}_i^V$ for the specific student i.

4.3 Answer prediction

With the attention weight w_t, the KVFKT model can predict the probability that a student answer q_t correctly by the following process. Different from

the traditional method, KVFKT integrates a three-parameter model of IRT during the prediction stage. It also considers not only students' abilities but also the difficulty coefficient and guess coefficient when predicting how students answer questions.

First, the KVFKT model reads the latent knowledge state from the value memory $\bar{M}_t^V$ after the second stage of forgetting, forming a read vector.

$$r_t = \sum_{i=1}^{N} w_{t,i}(\bar{M}_{t,i}^V)^T, \tag{8}$$

where $\bar{M}_{t,i}^V$ is the i-th row-vector of $\bar{M}_t^V$. Then, the read vector r_t and the question embedding vector Q_t are concatenated vertically and disseminated to a fully connected layer with the hyperbolic tangent activation so as to generate a feature vector f_t. This step can be expressed mathematically as follows:

$$f_t = tanh(W_f[r_t \oplus Q_t] + b_f). \tag{9}$$

Secondly, we utilize the acquired f_t to compute the corresponding student ability. Furthermore, we employ the question embedding vector Q_t to derive the relevant question difficulty coefficient. Additionally, the student's answer time at_{ij} is utilized to calculate the guessing coefficient. Subsequently, we will integrate these factors into the three-parameter IRT model, which accounts for the question difficulty and the guessing coefficient. The IRT model we use is as follows:

$$P(X_{ij} = 1|\theta_i; \beta_j, g_j) = g_j + (1-g_j)\frac{e^{D(\theta_i - \beta_j)}}{1 + e^{D(\theta_i - \beta_j)}}, \tag{10}$$

where θ_{tj} represents student ability, β_j represents question difficulty, g_j represents guess coefficient and D is a constant, we set it to 1.702 in accordance with the methodology employed in earlier studies (Lo, 2008). The calculation process of them is as follows:

$$\theta_{tj} = tanh(W_\theta f_t + b_\theta), \tag{11}$$

$$\beta_j = tanh(W_\beta Q_t + b_\beta), \tag{12}$$

$$g_j = Softmax(|\bar{at}_j - at_{i,j}| + \bar{g}_j), \tag{13}$$

where $\bar{at}_j$ represents the average time for answering question j, and at_{ij} represents the time for student i to answer question j. $\bar{g}_j$ represents the asymptotic value under the Item Characteristic Curve (ICC) (Von der Embse et al., 2018) of problem j. The student ability and question difficulty are parameterized by a weight matrix $\mathbf{W}$ and a bias vector $\mathbf{b}$ with appropriate dimensions, we also use the hyperbolic tangent to be the activation function for both networks such that both outputs are scaled into the range(-1,1). Finally, we feed each element into IRT and predict the students' response to the question:

$$P_t = g_j + (1-g_j)Softmax(3.0 * \theta_{tj} - \beta_j), \tag{14}$$

the output of the student ability network are multiplied by a factor of 3.0 for a practical reason(Yang and Kao, 2014).

4.4 Knowledge status update

After completing the prediction for the time node, the KVFKT model updates the value memory M_t^V based on the input tuple (q_t, y_t) and the attention weight w_t generated by the attention-getting stage. The KVFKT model first transforms (q_t, y_t) into ζ by the following rules:

$$\zeta = \begin{cases} q_t & y_t = 0 \\ q_t + n & y_t = 1 \end{cases}, \tag{15}$$

where n denotes the total number of questions included in the relevant datasets and y_t represents the real situation of students' answers to questions at time t. This approach enables us to differentiate between different representations of correct and incorrect responses.

Then KVFKT retrieves an embedding vector of ζ_t from a KC-response embedding matrix $\mathbf{B} \in \mathbb{R}^{2Q \times d_v}$. This embedding vector, denoted as $\mathbf{Z}_t \in \mathbb{R}^{d_v}$, represents the knowledge growth after working on the KC. When updating the memory, some of the memory is first erased with an erase vector $\mathbf{e}_t$ before new information is added to the memory with the add vector $a_t \in \mathbb{R}^{d_v}$. Erasing the memory offers the ability of forgetting similar to the KVFKT cell. Each value memory slot is updated as follows:

$$e_t = \sigma(W_e Z_t + b_e), \tag{16}$$

$$a_t = tanh(W_a Z_t + b_a), \tag{17}$$

$$M_{t+1,i}^V = \bar{M}_{t,i}^V \otimes (\mathbf{1} - w_{t,i} e_t)^T + w_{t,i} a_t^T. \tag{18}$$

To learn all parameters in KVFKT, we else choose the cross-entropy log loss between the prediction $\bar{y}_t$ and actual answer y_t as the objective function $\mathbb{L}(\theta) = -\sum_{t=1}^{N}(y_t log\bar{y}_t + (1 - y_t)log(1 - \bar{y}_t)) + \lambda_\theta \parallel \theta \parallel^2$, where θ denotes all learning parameters of KVFKT and λ_θ is the regularization hyper-parameter. The objective function was minimized using Adam optimizer on mini-batches.

5 Experiments

In this section, we introduce the dataset, the model baseline, and our experimental setups. Then, we conduct a large number of experiments to compare the performance of KVFKT with other KT models and the interpretability of KVFKT, to answer the following questions:

- **RQ1** Does KVFKT perform better at predicting student performance han other knowledge tracing modeling works?
- **RQ2** How do the different components designed in KVFKT affect the performance of KVFKT?
- **RQ3** Do the question difficulty, student ability, and guessing coefficients in the KVFKT prediction procedure help to explain the prediction results?
- **RQ4** How do the settings of forget cycle parameters influence the predictive performance of KVFKT?

5.1 Datasets and Baseline methods

We evaluate our model with four real-world datasets, the details are shown in Table 1. (1) **ASSISTments2012**[1]. This is the ASSISTments data for the school year 2012 2013 with affect predictions. (2) **EdNet**[2]. This dataset is a large-scale hierarchical student activity data set collected by Santa (an artificial intelligence guidance system). (3) **NeurIPS**[3] This dataset is from the Tasks 3 & 4 at the NeurIPS 2020 Education Challenge. (4) **FSAI-F1toF3**[4] This dataset is provided by the Find Solution AI Limited. We extracted the student interactions that are related to the mathematics curriculum from F.1 to F.3 in Hong Kong.

We compare KVFKT with several previous methods. For a fair comparison, all these methods are tuned to have the best performances. All models are trained on a cluster of Linux servers with RTX 3070 GPUs. The comparison methods include 7 models: **DKT (Piech et al., 2015)**, **DKVMN (Zhang et al., 2017)**, **AKT (Ghosh et al., 2020)**, **SAKT (Pandey and Karypis, 2019)** **SAINT (Choi et al., 2020)**, **LFBKT (Chen et al., 2022)**.

5.2 Parameter settings

We train the model using the Adam optimizer with a learning rate of 0.003 and a batch size of 32. To prevent gradient explosion, we set the norm clipping threshold to 10.0. Since the input sequences vary in length, we standardize all sequences to a length of 200. Sequences with fewer than 200 time steps are padded with zeros to fill the remaining steps. Masking is applied when computing the loss to account for the padding. Additionally, we set the forgetting cycle to 200,000, meaning that knowledge concepts (KCs) are fully forgotten after 200,000 time units. When initializing the forgetting matrix, we randomize it near the earliest time stamp to reflect the actual conditions of the dataset.

K-fold cross-validation is applied on the training set for hyperparameter selection across all datasets. For the FSAI-F1 to F3 dataset, due to its smaller size and to enhance generalization, we increase the value of K to 10. For the other datasets, we set K to 5. The network architectures of the DKT, DKVMN, AKT, SAKT, SAINT, and LFBKT models vary with different numbers of state dimensions and memory sizes. A grid search is conducted over the combinations of state dimension and memory size. The training and evaluation process is repeated five times (except for the FSAI-F1 to F3 dataset) to report model performance. We present the accuracy, root mean square error (RMSE), cross-entropy loss, and the average and standard deviation of the area under the ROC curve (AUC).

5.3 Student performance prediction(RQ1)

In KT, achieving student performance prediction accuracy often indicates that the model can accurately capture the student's knowledge state. To evaluate the model's performance, we use the area under the curve (AUC), accuracy (ACC), and root mean square error (RMSE) as metrics and compare them with five existing KT models. The corresponding results are presented in Table 2. Notably,

[1]https://sites.google.com/site/assistmentsdata/datasets/2012-13-school-data-with-affect

[2]https://github.com/riiid/ednet

[3]https://eedi.com/projects/neurips-education-challenge

[4]https://www.4littletrees.com/

$Dataset$	$N_{students}$	N_{skills}	$N_{questions}$	$Attempts\ Per\ skill$	$N_{sequence}$	$Rate_{Positive}$
ASSISTments2012	27,405	265	47,104	7,045	93.45	69.60%
EdNet	5,000	188	13,169	17	89.41	59.69%
FSAI-F1toF3	310	99	2,266	83	165.42	46.69%
NeurIPS	4,918	57	948	10	71.56	65.35%

Table 1: The summary of datasets.

Methods	ASSISTments2012			EdNet			FSAI-F1toF3			NeurIPS		
	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE
DKT	0.7349	0.7350	0.4434	0.7025	0.6666	0.4678	0.6942	0.6411	0.4885	0.7709	0.7045	0.4431
DKVMN	0.7226	0.7327	0.4485	0.6808	0.6579	0.4962	0.6840	0.6340	0.4952	0.7636	0.6981	0.4291
SAKT	0.6899	0.7250	0.4911	0.6982	0.6655	0.4861	0.7256	0.6953	0.4559	0.7474	0.6843	0.4414
AKT	0.7698	0.7440	0.4231	0.7217	0.6792	0.4687	0.6636	0.6215	0.4996	0.7721	0.7103	0.4168
SAINT	0.6713	0.7148	0.4712	0.7240	0.6786	0.4582	0.5997	0.5896	0.5121	**0.7841**	**0.7149**	**0.4144**
LFBKT	0.7955	0.7651	0.4605	0.6849	0.6528	0.4907	0.6523	0.6152	0.4810	0.7759	0.7134	0.4092
KVFKT	**0.8347**	**0.8066**	**0.4146**	**0.7344**	**0.6882**	**0.4476**	**0.7306**	**0.6981**	**0.4486**	0.7784	0.7108	0.4205

Table 2: Results of comparison methods on student performance prediction. KVFKT outperforms all baselines on all datasets.

Methods	ASSISTments2012			EdNet		
	AUC	ACC	RMSE	AUC	ACC	RMSE
w/o F	0.764	0.774	0.435	0.712	0.660	0.475
w/o G	0.807	0.785	0.428	0.725	0.685	0.465
w/o D	0.764	0.765	0.449	0.705	0.654	0.462
w/o IRT	0.754	0.745	0.458	0.695	0.645	0.480
KVFKT	**0.834**	**0.806**	**0.414**	**0.734**	**0.688**	**0.447**

Table 3: Results of ablation study.

KVFKT outperforms all other deep-learning-based KT methods on all datasets and metrics. Particularly on the ASSISTments2012 dataset, KVFKT surpasses the basic model by an average AUC improvement of 9.5% and outperforms the state-of-the-art SAINT model by enhancing ACC by 4.16%. This indicates that KVFKT is capable of incorporating these components into the model and capturing the learning gains.

5.4 Ablation study(RQ2)

In this subsection, we conduct several ablation experiments to assess the impact of each module on the model's final prediction outcomes. For these experiments, we focus on the best-performing datasets, ASSISTments2012 and EdNet, to highlight the results. As shown in Table 3, the first experiment removes the forgetting matrix, which significantly lowers the performance across all datasets. This result underscores the importance of accounting for forgotten information in the knowledge tracing model, making it more suited to the specific features of the dataset, thereby improving generalization and accuracy.

Second, when the guessing coefficient is removed from KVFKT, the overall impact on performance is somewhat reduced. This suggests that while the guessing coefficient influences accuracy, its effect is less significant compared to other factors. In the following experiment, we eliminate the difficulty factor from KVFKT. The performance of all datasets drops substantially compared to the KVFKT model, highlighting the critical role of the difficulty factor.

Finally, we remove the 3PL module from the prediction process, relying solely on the students' ability to answer questions. The resulting performance decline in both datasets emphasizes the importance of reintroducing the 3PL module to the knowledge tracing model, as it significantly enhances overall predictive accuracy.

5.5 Interpretability analysis(RQ3)

In this section, we demonstrate that the KVFKT model effectively captures the evolving knowledge states of students across multiple concepts during their learning process. To illustrate this, we randomly select a student from the ASSIST2012 dataset and track the transitions in the student's ability level, guessing coefficient, and the probability of correctly answering the next knowledge component (KC) as they progress through their learning. This is visualized using the first 30 attempts of the

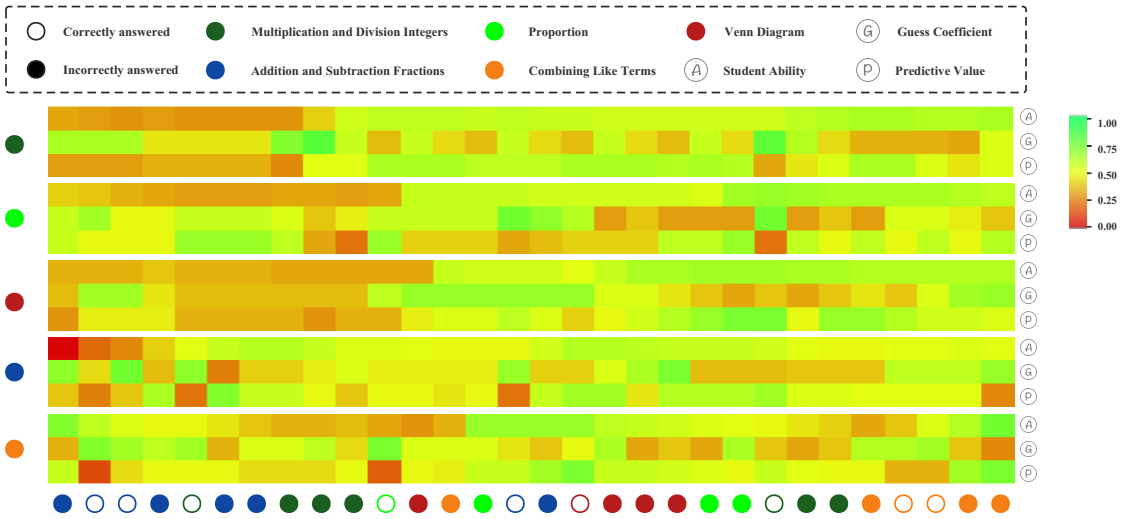

Figure 3: An example of a student's learning trajectory from the ASSIST2012 dataset is shown. The labels on the vertical axis correspond to different skill tags. The learning trajectory is represented along the horizontal axis using filled and hollow circles, with corresponding colors. Filled circles indicate a correct response, while hollow circles represent an incorrect response. The learning trajectory of each student is depicted by three heatmaps: 'Student Ability' (A), 'Guess Coefficient' (G), and 'Predictive Value' (P). The values in these heatmaps range from 0 to 1.

student.

As shown in Figure 3, the transition of students' abilities is intuitive. When relevant KCs are introduced, we observe a noticeable improvement in the student's knowledge within those domains. In contrast, the abilities related to KCs that are not currently being tested decline, primarily due to the effects of forgetting. For instance, the "Combining Like Terms" KC experiences a continuous decline in the first 10 steps due to insufficient review. Additionally, the transition of the guessing coefficient is less smooth, reflecting its dependence on the time spent by students on each problem. The combination of a student's ability and the guessing coefficient jointly influences the prediction, which may lead to discrepancies between the predicted and actual ability levels. For example, although the student shows clear mastery in the "Addition and Subtraction Fractions" KC, an excessively high guessing coefficient in the fifth step introduces uncertainty in the prediction.

Moreover, due to the complexity of the KVFKT model and its numerous parameters and features, some anomalies arise that are difficult to explain. For instance, in the fourth phase, KVFKT erroneously predicts both a low ability and a low guessing coefficient as a correct response.

Finally, we observe an intriguing phenomenon: the student's performance on the "Venn Diagram" activities did not decline when the "Addition and Subtraction Fractions" questions appeared. This unexpected trend may suggest a potential connection between the "Venn Diagram" KC and the "Addition and Subtraction Fractions" KC for this particular student. This observation raises interesting questions about how relationships between different KCs could influence learning, pointing to the need for further research on improving the reliability of deep learning-based knowledge tracing models.

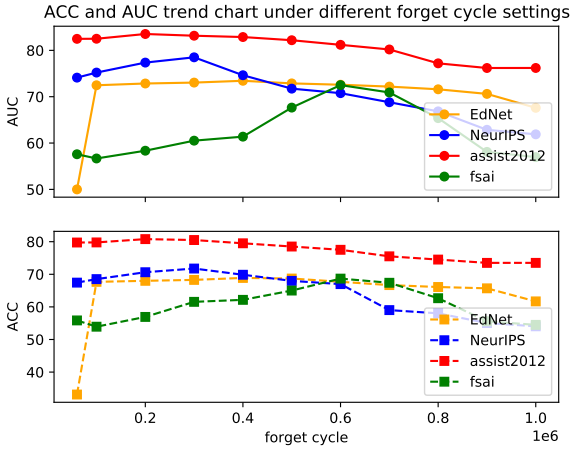

Figure 4: The impact of forget cycle hyperparameters on experimental results for each dataset.

5.6 Experiment summary on forget cycle hyperparameters(RQ4)

To investigate KVFKT's sensitivity to the forgetting phenomenon, this section explores how adjusting the $forget_{cycle}$ parameter affects the model's

predictive outcomes. The $forget_{cycle}$ parameter represents the time interval (in timestamp units) required for a student to completely forget a knowledge point after mastering it. This parameter ranges from 60,000 to 1,000,000 timestamps, which corresponds to 16.6 hours to 277 hours. As shown in Figure 4, the experimental results demonstrate a trend where the AUC and ACC values initially increase and then decrease as the $forget_{cycle}$ parameter is adjusted. The peak values also vary across different datasets. Based on the experimental findings, we draw the following conclusions:

(1) KVFKT not only functions as a knowledge tracing model but also has the capability to estimate the average forgetting cycle for students. Students from different datasets exhibit varying forgetting cycles. For instance, students in the ASSISTments2012 dataset tend to forget mastered knowledge points within approximately 55 hours, while students in the EdNet dataset typically show forgetting cycles of around 110 hours.

(2) Modeling forgetting behavior proves to be crucial for KVFKT, with experimental results validating its significance. Different forgetting cycles lead to markedly different predictive outcomes, underscoring the importance of the $forget_{cycle}$ parameter in addressing knowledge tracing problems and its role in refining the model's performance.

6 Conclusion

In this paper, we propose a novel model called KVFKT. To enhance the interpretability of deep learning models, we incorporate the complex forgetting phenomenon and integrate the IRT three-parameter model. Through extensive experiments on four public datasets, we demonstrate that KVFKT is capable of capturing more realistic and meaningful knowledge state evolutions throughout the learning process of students.

7 Discussion and Future Work

Limitations. Firstly, the current KVFKT model does not fully account for the vast range of individual differences in student learning. Secondly, another significant limitation lies in how the KVFKT model treats the relationships between different KCs. Thirdly, as the model incorporates more parameters, such as the item discrimination parameter and dynamic forgetting adjustments, there is a risk that the model could become too complex to scale efficiently.

Future work. In future research, the following directions will be explored. First, developing adaptive mechanisms to better capture the diversity of student learning behaviors. This includes integrating personalized learning features based on demographic, cognitive, and engagement data to enhance the precision of the model. Second, investigating advanced techniques, such as graph-based or neural relational models, to more accurately represent and leverage the interdependencies among different KCs. This will facilitate a deeper understanding of the dynamics of knowledge structures. Third, designing and evaluating lightweight parameter optimization methods to ensure that the inclusion of additional parameters, such as item discrimination and dynamic forgetting adjustments, does not hinder scalability. Potential approaches may involve techniques such as parameter pruning, parallel computing, or efficient approximation algorithms.

Acknowledgments

This work was supported by the National Key R&D Program of China (2022YFC3303603), National Natural Science Foundation of China (62077028, 62377028, 62276114), Guangdong Basic and Applied Basic Research Foundation (2023B1515120064), Key Laboratory of Smart Education of Guangdong Higher Education Institutes, Jinan University (2022LSYS003), Guangdong-Macao Advanced Intelligent Computing Joint Laboratory (2020B1212030003), the Fundamental Research Funds for the Central Universities, JNU (21623202), Doctoral Program for Cultivating Top-notch Innovative Talents, Jinan University (2024CXB037), and High Performance Computing Platform of South China University of Technology.

References

Ghodai Abdelrahman and Qing Wang. 2022. Deep graph memory networks for forgetting-robust knowledge tracing. *IEEE Transactions on Knowledge and Data Engineering*.

Ghodai Abdelrahman, Qing Wang, and Bernardo Nunes. 2023. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11):1–37.

Lee Averell and Andrew Heathcote. 2011. The form of the forgetting curve and the fate of memories. *Journal of mathematical psychology*, 55(1):25–35.

Yanhong Bai, Jiabao Zhao, Tingjiang Wei, Qing Cai,

and Liang He. 2024. A survey of explainable knowledge tracing. *Applied Intelligence*, pages 1–32.

Mingzhi Chen, Quanlong Guan, Yizhou He, Zhenyu He, Liangda Fang, and Weiqi Luo. 2022. Knowledge tracing model with learning and forgetting behavior. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM-2022)*, pages 3863–3867.

Benoît Choffin, Fabrice Popineau, Yolaine Bourda, and Jill-Jênn Vie. 2019. Das3h: Modeling student learning and forgetting for optimally scheduling distributed practice of skills. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM-2019)*.

Youngduck Choi, Youngnam Lee, Junghyun Cho, Jineon Baek, Byungsoo Kim, Yeongmin Cha, Dongmin Shin, Chan Bae, and Jaewe Heo. 2020. Towards an appropriate query, key, and value computation for knowledge tracing. In *Proceedings of the Seventh ACM Conference on Learning@ Scale (L@S-2020)*, pages 341–344.

Albert T Corbett and John R Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278.

Albert T. Corbett and John R. Anderson. 2005. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4:253–278.

Jiajun Cui, Hong Qian, Bo Jiang, and Wei Zhang. 2024. Leveraging pedagogical theories to understand student learning process with graph-based reasonable knowledge tracing. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD-2024)*, pages 502–513.

Xiuliang Duan, Dating Tan, Liangda Fang, Yuyu Zhou, Chaobo He, Ziliang Chen, Lusheng Wu, Guanliang Chen, Zhiguo Gong, Weiqi Luo, and Quanlong Guan. 2024. Reason-and-execute prompting: Enhancing multi-modal large language models for solving geometry questions. In *Proceedings of the 32nd ACM International Conference on Multimedia (MM-2024)*, pages 6959–6968.

Hermann Ebbinghaus. 1885. *Über das gedächtnis: untersuchungen zur experimentellen psychologie*.

Aritra Ghosh, Neil Heffernan, and Andrew S Lan. 2020. Context-aware attentive knowledge tracing. In *The 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2020)*, pages 2330–2339.

Quanlong Guan, Xinghe Cheng, Fang Xiao, Zhuzhou Li, Chaobo He, Liangda Fang, Guanliang Chen, Zhiguo Gong, and Weiqi Luo. 2025. Explainable exercise recommendation with knowledge graph. *Neural Networks*.

Xiaoyu Han, Shu Zhang, Juxiang Zhou, Zijie Li, and Jun Wang. 2023. Deep knowledge tracing with gru and learning state enhancement. In *Machine Learning for Cyber Security: 4th International Conference (ML4CS-2022)*, pages 677–686.

Chaobo He, Hao Cheng, Jiaqi Yang, Yong Tang, and Quanlong Guan. 2025. Signed graph embedding via multi-order neighborhood feature fusion and contrastive learning. *Neural Networks*, 182.

Zhenya Huang, Qi Liu, Yuying Chen, Le Wu, Keli Xiao, Enhong Chen, Haiping Ma, and Guoping Hu. 2020. Learning or forgetting? a dynamic approach for tracking the knowledge proficiency of students. *ACM Transactions on Information Systems (TOIS)*, 38(2):1–33.

Mohammad M Khajah, Yun Huang, José P González-Brenes, Michael C Mozer, and Peter Brusilovsky. 2014. Integrating knowledge tracing and item response theory: A tale of two frameworks. In *Workshop Proceedings of the 22nd Conference on User Modeling, Adaptation, and Personalization*, volume 1181, pages 7–15.

Unggi Lee, Sungjun Yoon, Joon Seo Yun, Kyoungsoo Park, YoungHoon Jung, Damji Stratton, and Hyeoncheol Kim. 2023. Difficulty-focused contrastive learning for knowledge tracing with a large language model-based difficulty prediction. *arXiv preprint arXiv:2312.11890*.

Sheng Li, Quanlong Guan, Liangda Fang, Fang Xiao, Zhenyu He, Yizhou He, and Weiqi Luo. 2022. Cognitive diagnosis focusing on knowledge concepts. In *Proceedings of the 31st ACM International Conference on Information and Knowledge Management (CIKM-2022)*, page 3272–3281.

Guimei Liu, Huijing Zhan, and Jung-jae Kim. 2024. Question difficulty consistent knowledge tracing. In *Proceedings of the ACM on Web Conference 2024 (WWW-2024)*, pages 4239–4248.

Qi Liu, Shiwei Tong, Chuanren Liu, Hongke Zhao, Enhong Chen, Haiping Ma, and Shijin Wang. 2019. Exploiting cognitive structure for adaptive learning. In *The 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2019)*, pages 627–635.

Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, Boyu Gao, Weiqi Luo, and Jian Weng. 2023. Enhancing deep knowledge tracing with auxiliary tasks. In *Proceedings of the ACM Web Conference 2023 (WWW-2023)*, pages 4178–4187.

Shih-Ching Lo. 2008. Equal area estimation of three-parameter logistic model for item response theory. In *Proceedings of the 3rd IEEE Asia-Pacific Services Computing Conference (APSCC-2008)*, pages 1447–1452.

Haiping Ma, Yong Yang, Chuan Qin, Xiaoshan Yu, Shangshang Yang, Xingyi Zhang, and Hengshu Zhu.

2024. Hd-kt: Advancing robust knowledge tracing via anomalous learning interaction detection. In *Proceedings of the ACM on Web Conference 2024 (WWW-2024)*, pages 4479–4488.

Shaul Markovitch and Paul D Scott. 1988. The role of forgetting in learning. In *Proceedings of the Fifth International Conference on Machine Learning (ML-1988)*, pages 459–465.

Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. 2019. Augmenting knowledge tracing by considering forgetting behavior. In *The World Wide Web Conference (WWW-2019)*, pages 3101–3107.

Qin Ni, Tingjiang Wei, Jiabao Zhao, Liang He, and Chanjin Zheng. 2023. Hhskt: A learner–question interactions based heterogeneous graph neural network model for knowledge tracing. *Expert Systems with Applications*, 215:119334.

Shalini Pandey and George Karypis. 2019. A self attentive model for knowledge tracing. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM-2019)*.

Shalini Pandey and Jaideep Srivastava. 2020. Rkt: Relation-aware self-attention for knowledge tracing. In *The 29th ACM International Conference on Information and Knowledge Management (CIKM-2020)*, pages 1205–1214.

Zachary A Pardos and Neil T Heffernan. 2010. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *User Modeling, Adaptation, and Personalization: 18th International Conference (UMAP-2010)*, pages 255–266.

Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. *Advances in Neural Information Processing Systems (NIPS-2015)*, 28.

Georg Rasch. 1993. *Probabilistic models for some intelligence and attainment tests*. ERIC.

Timothy J Ricker, Evie Vergauwe, and Nelson Cowan. 2016. Decay theory of immediate memory: From brown (1958) to today (2014). *Quarterly Journal of Experimental Psychology*, 69(10):1969–1995.

Tomás J Ryan and Paul W Frankland. 2022. Forgetting as a form of adaptive engram cell plasticity. *Nature Reviews Neuroscience*, 23(3):173–186.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems (NIPS-2017)*, 30.

Nathaniel Von der Embse, Dane Jester, Devlina Roy, and James Post. 2018. Test anxiety effects, predictors, and correlates: A 30-year meta-analytic review. *Journal of Affective Disorders*, 227:483–493.

Frances M Yang and Solon T. Kao. 2014. Item response theory for measurement validity. *Shanghai Archives of Psychiatry*, 26:171 – 177.

Michael V. Yudelson, K. Koedinger, and Geoffrey J. Gordon. 2013. Individualized bayesian knowledge tracing models. In *Proceedings of the 18th International Conference on Artificial Intelligence in Education (AIED-2013)*.

Jiani Zhang, Xingjian Shi, and King. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web (WWW-2017)*, pages 765–774.

Weizhong Zhao, Jun Xia, Xingpeng Jiang, and Tingting He. 2023. A novel framework for deep knowledge tracing via gating-controlled forgetting and learning mechanisms. *Information Processing and Management*, 60(1):103114.

Hanqi Zhou, Robert Bamler, Charley M Wu, and Álvaro Tejero-Cantero. 2024. Predictive, scalable and interpretable knowledge tracing on structured domains. *arXiv preprint arXiv:2403.13179*.