# Rethinking and Improving Student Learning and Forgetting Processes for Attention Based Knowledge Tracing Models

**Youheng Bai**[1], **Xueyi Li**[1], **Zitao Liu**[1*], **Yaying Huang**[1], **Mi Tian**[2], **Weiqi Luo**[1]

[1]Guangdong Institute of Smart Education, Jinan University, Guangzhou, Guangdong, 510632, China
[2]TAL Education Group, Beijing, 102206, China
baiyouheng@outlook.com, lixueyi@stu2021.jnu.edu.cn, {liuzitao, huangyaying, lwq}@jnu.edu.cn, tianmi@tal.com

## Abstract

Knowledge tracing (KT) models students' knowledge states and predicts their future performance based on their historical interaction data. However, attention based KT models struggle to accurately capture diverse forgetting behaviors in ever-growing interaction sequences. First, existing models use uniform time decay matrices, conflating forgetting representations with problem relevance. Second, the fixed-length window prediction paradigm fails to model continuous forgetting processes in expanding sequences. To address these challenges, this paper introduces LefoKT, a unified architecture that enhances attention based KT models by incorporating proposed relative forgetting attention. LefoKT improves forgetting modeling through relative forgetting attention to decouple forgetting patterns from problem relevance. It also enhances attention based KT models' length extrapolation capability for capturing continuous forgetting processes in ever-growing interaction sequences. Extensive experimental results on three datasets validate the effectiveness of LefoKT.

**Code** — https://pykt.org/

## Introduction

Knowledge tracing (KT) aims to model and trace students' knowledge states by analyzing their historical interaction data to predict their performance on new questions (Corbett and Anderson 1994; Piech et al. 2015; Abdelrahman, Wang, and Nunes 2023), as illustrated in Figure 1. Such capabilities play a key role in facilitating personalized instruction and enhancing learning experiences, both of which are crucial to next-generation intelligent education systems (Shehata et al. 2023).

Recently, attention mechanisms have achieved significant success in natural language processing and graph analytics (Vaswani et al. 2017; Luo et al. 2024) due to their capability to capture long-term dependencies. This capability is crucial in the KT task and has contributed to the development of many attention based KT models. Furthermore, effectively modeling students' forgetting behaviors in educational settings is crucial for KT task, as it implicitly affects the estimation of student knowledge states. Some KT models leverage attention mechanisms to model learning and forgetting processes in student interaction sequences. For example, AKT (Ghosh, Heffernan, and Lan 2020) and RKT (Pandey and Srivastava 2020) combine attention mechanisms with forgetting modeling by incorporating time decay matrices into attention scores.

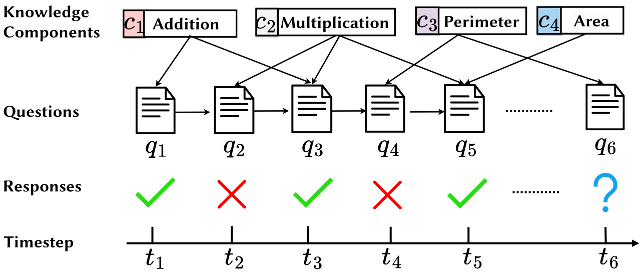


Figure 1: An illustration of the KT task.

Although these attention based KT models have made progress in capturing students' forgetting behavior, challenges remain in accurately modeling the complex learning and forgetting processes. First, using time-decay matrices to model students' forgetting processes oversimplifies complex forgetting patterns associated with different learning habits and conflates forgetting patterns with problem relevance. Second, existing KT models that use fixed-length windows during prediction truncate longer interaction sequences, which fails to trace students' knowledge forgetting over ever-growing learning sequences, as they lose crucial information about students' long-term knowledge retention and gradual forgetting over time.

Taking the aforementioned challenges into account, we propose the LefoKT framework to rethink the learning and forgetting processes in attention based KT models. LefoKT enhances the modeling of diverse forgetting behaviors and improves prediction performance for ever-growing learning sequences, enabling more accurate modeling of students' long-term forgetting processes. Specifically, LefoKT integrates relative forgetting attention (RFA) to capture chronological order and model multiple forgetting behaviors by leveraging various temporal decay biases of relative positional encodings (RPEs). Furthermore, RFA can improve the length extrapolation capability of LefoKT, which allows the

[*]Corresponding author.

model to trace students' knowledge states across different sizes of context windows in the attention mechanism (Press, Smith, and Lewis 2022; Qin, Zhong, and Deng 2024). We conduct experiments on three KT datasets, and the results demonstrate that LefoKT enhances the forgetting modeling capabilities of attention based KT models, maintaining stable prediction performance across varying sequence lengths.

Our contributions are three-fold:

- We introduce relative forgetting attention which effectively captures varying forgetting patterns by incorporating different RPEs into attention scores.
- We propose LefoKT, a flexible forgetting-aware framework that enhances learning and forgetting processes modeling through relative forgetting attention.
- We comprehensively assess the effectiveness of LefoKT framework in modeling student' learning and forgetting processes. Additionally, we reveal the implicit relationship between forgetting process and length extrapolation.

## Related Work

### Attention Based Knowledge Tracing

Attention mechanisms have transformed the KT task by aggregating information and capturing the dependencies in students' learning histories, leading to more accurate predictions of future performance (Li et al. 2024c). Pandey et al. were the first to introduce dot-product attention to extract knowledge states from students' past learning history (Pandey and Karypis 2019). Since then, many attention based KT models have been developed. Choi et al. used a Transformer based architecture to process a sequence of student interactions and responses (Choi et al. 2020). Ghosh et al. proposed a monotonic attention mechanism to model students' forgetting process (Ghosh, Heffernan, and Lan 2020). Liu et al. utilized question-specific variations to distinguish individual differences among questions testing the same knowledge sets (Liu et al. 2023). Yin et al. used a Transformer based model with a two-level framework to realize stable knowledge state estimation (Yin et al. 2023).

### Forgetting Modeling in Knowledge Tracing

Modeling student forgetting behaviors is crucial for accurate knowledge tracing, as it reflects the natural decay of knowledge over time and impacts prediction accuracy. Many studies have aimed to explore more effective ways to capture forgetting factors. Some KT models capture implicit features of students' forgetting behaviors from the time intervals of previous interactions and response time, such as DKT-Forget (Nagatani et al. 2019), SAINT (Choi et al. 2020), and HawkesKT (Wang et al. 2021). Another effective approach is to exploit the recurrent mechanism, which inherently consists of a forgetting gate, to capture forgetting features, such as LPKT (Shen et al. 2021) and LBKT (Li et al. 2024d). Furthermore, inspired by the Ebbinghaus forgetting curve in psychology, some researchers introduce exponential decay to model forgetting behaviors, such as AKT (Ghosh, Heffernan, and Lan 2020) and RKT (Pandey and Srivastava 2020).

### Relative Positional Encodings

Positional encodings (PEs) play a crucial role in incorporating temporal information into sequence modeling tasks. Among various PEs, RPEs have emerged as the mainstream approach due to their flexibility and effectiveness in capturing sequence order (Kazemnejad et al. 2023). Various RPE methods have been proposed in recent literature. T5 bias integrates no explicit position information into the self-attention value vectors by adding a shared learnable bias to query-key scores (Raffel et al. 2020). ALiBi provides positional information by applying a linear penalty to query-key attention scores, proportional to their relative distances (Press, Smith, and Lewis 2022). KERPLE uses shift-invariant conditionally positive definite kernels, including power-based and logarithmic-based variants, to model relative position differences (Chi et al. 2022). FIRE enhances Transformer extrapolation to longer sequences by using a learnable mapping function and a progressive interpolation technique that adapts to sequence length (Li et al. 2024a). Sandwich applies a simplified version of sinusoidal positional encoding to create a log-decaying temporal bias pattern in attention scores (Chi et al. 2023).

## Methodology

In this section, we first describe the KT task. We then rethink forgetting modeling in attention based KT models and propose a novel relative forgetting attention via RPEs. Finally, we introduce the components of LefoKT framework.

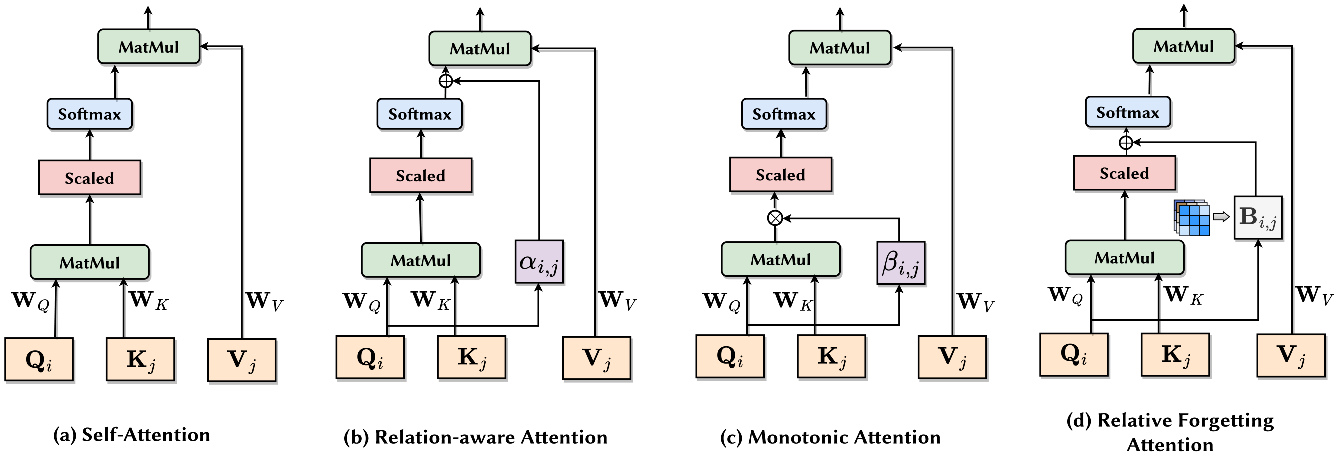
### Problem Definition

The objective of KT tasks is to predict the probability of a student correctly answering an upcoming question based on their historical exercise sequences. Each historical sequence, denoted as $X_t = \langle x_1, x_2, \ldots, x_t \rangle$, consists of interactions $x_i = (q_i, \{c\}, r_i, t_i)$, where $q_i$ is the question ID, $\{c\}$ is the set of knowledge components (KCs) related to question $q_i$, $r_i$ is the response correctness (1 for correct, 0 for incorrect), and $t_i$ is the time step of the response. A KC is a description of a mental structure or process that a learner uses, alone or in combination with other KCs, to accomplish steps in a task or a problem[1]. We estimate probability $\hat{r}_{t+1} = p\left(r_{t+1} = 1 \mid X_t, q_{t+1}\right)$, where $\hat{r}_{t+1}$ is the likelihood of correctly answering question $q_{t+1}$.

### Rethinking Forgetting Modeling in Attention Based Knowledge Tracing Models

**Revisiting Self-attention** Self-attention mechanisms in KT enable models to focus on relevant historical interaction within input sequences, representing student interactions with questions and KCs. As shown in Figure 2(a), these KT models use attention weights to measure the importance of past interactions for predicting future performance. Formally, the input interactions are represented by matrices $\mathbf{Q}, \mathbf{K}, \mathbf{V} \in \mathbb{R}^{n \times d}$, where $n$ is the interaction length and $d$ is the feature dimension. In a multi-head attention setup, these

[1] A KC is a generalization of everyday terms like concept, principle, fact, or skill.



(a) Self-Attention

(b) Relation-aware Attention

(c) Monotonic Attention

(d) Relative Forgetting Attention

Figure 2: Overview of the forgetting modeling in attention based KT models and RFA module. For simplicity, only single-head operations are illustrated.

matrices are projected into $h$ different subspaces, where for each head $l$, the attention mechanism computes the attention scores and produces the student knowledge state representation as follows:

$$\mathrm{A}(\mathbf{Q}_i^{(l)}, \mathbf{K}_j^{(l)}) = \mathrm{SF}((\mathbf{Q}_i^{(l)}\mathbf{W}_Q^{(l)})(\mathbf{K}_j^{(l)}\mathbf{W}_K^{(l)})^T/\sqrt{d_k})$$

$$\mathbf{O} = \|_{l=1}^{h} \left(\mathrm{A}(\mathbf{Q}_i^{(l)}, \mathbf{K}_j^{(l)})\mathbf{V}_j^{(l)}\mathbf{W}_V^{(l)}\right) \quad (1)$$

where A denotes the attention function for the $l$-th head, SF is the softmax operation, and $\mathbf{Q}_i^{(l)}$, $\mathbf{K}_j^{(l)}$, $\mathbf{V}_j^{(l)}$ are the query, key, and value vectors for the $l$-th head, with $d_k = d/h$ being the scaling factor. The trainable weight matrices are $\mathbf{W}_Q^{(l)}$, $\mathbf{W}_K^{(l)}$, $\mathbf{W}_V^{(l)} \in \mathbb{R}^{d_k \times d_k}$. In eq.(1), $\|$ concatenates the outputs from $h$ heads to form the final representation $\mathbf{O} \in \mathbb{R}^{n \times d}$.

**Relative Forgetting Attention** Attention based KT models have advanced in integrating forgetting mechanisms. However, some models like RKT (Pandey and Srivastava 2020) and AKT (Ghosh, Heffernan, and Lan 2020) still exhibit limitations. These KT models often conflate question-related factors with forgetting behaviors and rely on fixed exponential decay functions to model forgetting processes, which may not fully capture complex forgetting patterns in students' learning processes.

To capture complex forgetting patterns, we propose an RFA module, with an overall implementation illustrated in Figure 2(d). It decouples the modeling of forgetting from question-specific factors by incorporating flexible temporal decaying biases via RPEs. RFA enables the KT model to adapt to a variety of forgetting patterns, including linear and exponential decay; reflecting differences in students' forgetting behaviors depending on their learning habits and contexts (Bailey 1989; Fisher and Radvansky 2022). Specifically, RFA implements a distance-aware relative forgetting attention score RFA for $l$-th head, which is defined as:

$$\mathrm{RFA}(\mathbf{Q}_i^{(l)}, \mathbf{K}_j^{(l)}) = \mathrm{SF}\left(\frac{(\mathbf{Q}_i^{(l)}\mathbf{W}_Q^{(l)})(\mathbf{K}_j^{(l)}\mathbf{W}_K^{(l)})^T}{\sqrt{d_k}} + \mathbf{B}_{i,j}^{(l)}\right)$$

$$\mathbf{B}_{i,j}^{(l)} = \begin{cases} \mathcal{F}(i,j), & \text{if } i \geq j \\ -\infty, & \text{otherwise} \end{cases}$$

where $\mathbf{B}_{i,j}^{(l)}$ is the causal distance decay matrix that models forgetting behavior, derived from the RPE function $\mathcal{F}(i,j)$. We implement six different RPEs as specific forms of $\mathcal{F}(i,j)$, summarized in Table 1. These RPEs model forgetting by relative distance and further adjust the decay rate of the attention scores to capture various forgetting patterns, allowing the model to adapt to different forgetting behaviors.

| RPEs | $\mathcal{F}(i,j)$ |
|---|---|
| T5 Bias | $b_{bucket}(i-j)$ |
| ALiBi | $\frac{1}{2^{h/2}} \cdot \lvert i-j \rvert$ |
| KERPLE-Log | $-r_1 \log(1 + r_2 \lvert i-j \rvert)$ |
| KERPLE-Power | $-r_1 \lvert i-j \rvert^{r_2}$ |
| Sandwich | $r_1 \sum_{k=1}^{r_2} \cos\left(\frac{(i-j)}{10000^{k/r_2}}\right)$ |
| FIRE | $f_\theta\left(\frac{\psi(i-j)}{\psi(\max\{L,i\})}\right)$ |

Table 1: Implementations of $\mathcal{F}(i,j)$ by different RPEs, where $i$ and $j$ represent the query and key indexes.

First, RFA incorporates a temporal decay bias $\mathbf{B}_{i,j}^{(l)}$, implemented through RPEs $\mathcal{F}(i,j)$ that capture relative distances between interactions. By utilizing these position-aware decay biases, RFA adapts to various temporal relationships in the sequence, thereby capturing diverse forgetting patterns. Furthermore, RFA can adjust various forgetting rates simultaneously through multi-head mechanism. Figure 3(a) demonstrates how different RPEs model forgetting patterns: the orange curve represents exponential forgetting, while the blue curve shows linear forgetting. The

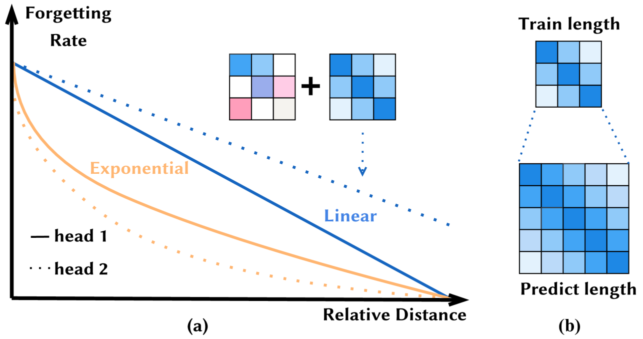


Figure 3: (a) RFA for forgetting modeling; (b) Comparison of decay patterns in KT length extrapolation.

solid and dotted lines for each curve illustrate how different attention heads adjust these forgetting rates. This multi-head approach enables RFA to adapt to diverse forgetting behaviors observed in real-world learning scenarios. To incorporate these different forgetting patterns into the attention mechanism, RFA combines decay bias matrices with attention score matrices. The final output of RFA is then computed by aggregating the outputs from all attention heads:

$$\mathbf{O}_{\text{RFA}} = \|_{l=1}^{h} \left( \text{RFA}(\mathbf{Q}_i^{(l)}, \mathbf{K}_j^{(l)}) \mathbf{V}_j^{(l)} \mathbf{W}_V^{(l)} \right) \quad (2)$$

where $\mathbf{O}_{\text{RFA}}$ denotes the interaction representation after forgetting modeling.

In addition to forgetting modeling abilities, the RFA module also enhances length extrapolation for KT models across ever-growing sequence lengths. Traditional KT models are trained on fixed-length truncated windows and inevitably lose critical information about long-term forgetting, limiting their ability to model students' extended forgetting behaviors in longer sequences. By leveraging RPEs, the decay biases ensure that the decay rates generated remain consistent regardless of prediction sequence length, as they depend solely on relative distances. As illustrated in Figure 3(b), decay biases exhibit identical attenuation trends for both short sequences used in training and longer sequences encountered during prediction. This allows the KT model to maintain uniform forgetting dynamics across varying prediction lengths, enabling accurate predictions for extended interaction histories. RFA improves KT model's ability to model long-term forgetting in real-world educational scenarios, where interaction sequences are growing and have inconsistent length distributions.

Please note that while attention based KT models that focus on forgetting, such as AKT and RKT, use a temporal decay to capture forgetting (as shown in Figure 2 (b) and (c)), they tend to conflate problem-relevant factors with forgetting behaviors and rely on exponential decay pattern. These approaches may oversimplify the complexity of forgetting patterns in the learning process and are often limited to capturing forgetting within fixed-length windows, struggling to adapt to varying sequence lengths. In contrast, as shown in Figure 2(d), the RFA module separates forgetting from problem-specific factors via RPEs, which more accurately captures a wide range of forgetting patterns while enabling effective prediction of ever-growing interaction sequences.

## The LefoKT Framework

The architecture of the LefoKT framework is composed of the input representation, RFA module, point-wise feed-forward network and prediction layer.

**Input Representation** In educational scenarios, the number of student interactions (such as responses to questions) typically far exceeds the number of KCs. Therefore, we adopt a method that converts the question-response format into a KC-response format. This transformation allows for efficient learning and fair comparisons for sparse problem-response KT data. We use an embedding layer to obtain representations of questions $\mathbf{E} \in \mathbb{R}^{n \times d}$ and interactions $\mathbf{I} \in \mathbb{R}^{n \times d}$. For the $t$-th sequence, $\mathbf{E}_t$ and $\mathbf{I}_t$ are represented as:

$$\mathbf{E}_t = \text{Enc}^Q(c_t, q_t); \quad \mathbf{I}_t = \text{Enc}^I(c_t, q_t, r_t)$$

where $\text{Enc}^Q$ and $\text{Enc}^I$ are encoding functions for questions and interactions respectively.

**RFA Module** To model interaction relations and forgetting behaviors in a sequence of student interactions, we introduce RFA attention into the LefoKT framework. The RFA module captures dependencies in the student's continuous learning interactions while modeling multiple forgetting behaviors over time. The forgetting-aware attention output $\mathbf{O}_{\text{RFA}} \in \mathbb{R}^{t \times d}$ is computed based on the RFA mechanism defined in eq.(2), with the following input matrices:

$$\mathbf{Q} = \mathbf{E}_{1:t}; \ \mathbf{K} \in \{\mathbf{E}_{1:t}, \mathbf{I}_{1:t}\}; \ \mathbf{V} = \mathbf{I}_{1:t}$$

where query and value matrices $\mathbf{Q}, \mathbf{V} \in \mathbb{R}^{t \times d}$ are formed from the question embeddings and interaction representations respectively, while the key matrix $\mathbf{K} \in \mathbb{R}^{t \times d}$ can be constructed from either representation. Here, $t$ denotes the length of the interaction sequence, $\mathbf{E}_{1:t} = [\mathbf{E}_1, \cdots, \mathbf{E}_t]$ and $\mathbf{I}_{1:t} = [\mathbf{I}_1, \cdots, \mathbf{I}_t]$ denote question and interaction representations, respectively.

**Point-wise Feed-Forward Network** In the KT task, effectively modeling the intricate relationships within student interaction sequences is essential. To enhance this capability, we incorporate a point-wise feed-forward network following the RFA output. This layer introduces non-linearity to the model, allowing it to capture more sophisticated patterns in student learning sequences. It transforms the RFA module output through a series of operations:

$$\tilde{\mathbf{O}} = (\text{Dropout}\,(\phi\,(\mathbf{O}_{\text{RFA}}\mathbf{W}_1 + \mathbf{b}_1))\,\mathbf{W}_2 + \mathbf{b}_2)$$

$$\mathbf{H} = \text{LayerNorm}\left(\mathbf{O}_{\text{RFA}} + \text{Dropout}\left(\tilde{\mathbf{O}}\right)\right)$$

where $\mathbf{W}_1, \mathbf{W}_2 \in \mathbb{R}^{d \times d}$ and $\mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}^d$ are learnable parameters, $\phi$ is ReLU function, and $\mathbf{H}$ is the final interaction output.

**Prediction Layer** After obtaining the output, the knowledge state $\mathbf{H}_{t+1}$ for the next question $\mathbf{Q}_{t+1}$ at time step $t+1$ is calculated. Subsequently, the final predicted response $\hat{r}_{t+1}$ is generated by the prediction layer. We use a two-layer fully connected network to capture the complex nonlinear relationship between the output representations. The network is optimized by minimizing the binary cross-entropy loss between the actual response $r_{t+1}$ and the predicted response $\hat{r}_{t+1}$. We can represent the process as follows:

$$\hat{r}_{t+1} = \sigma\left(\phi\left(\mathbf{W}_2 \cdot \phi\left(\mathbf{W}_1 \cdot [\mathbf{H}_{t+1}; \mathbf{Q}_{t+1}] + \mathbf{b}_1\right) + \mathbf{b}_2\right)\right)$$

$$\mathcal{L} = -\sum_{t}\left(r_{t+1} \cdot \log \hat{r}_{t+1} + (1 - r_{t+1}) \cdot \log\left(1 - \hat{r}_{t+1}\right)\right)$$

where $\sigma$ denotes Sigmoid function. $\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2$ are trainable parameters and $\mathcal{L}$ represents the binary cross-entropy loss function.

# Experiments

## Datasets

We selected three widely used benchmark datasets to evaluate the performance of LefoKT model. The dataset statistics are summarized in Table 2. These datasets are as follows:

- Algebra 2005-2006 (**AL2005**): The AL2005 dataset is sourced from the KDD Cup 2010 EDM Challenge, which features interactions of 13-14 year-old students with Algebra questions. It provides detailed step-level responses from students to mathematical problems (Stamper and Pardos 2016). In our experiment, we used the concatenation of the problem name and step name to create a unique identifier for each question.
- Bridge to Algebra 2006-2007 (**BD2006**): The BD2006 dataset contains mathematical problems from the interaction logs of students using intelligent tutoring systems (Stamper and Pardos 2016). We used the same method as AL2005 to construct unique question identifiers.
- NeurIPS2020 Education Challenge (**NIPS34**): The NIPS34 dataset is provided by the NeurIPS 2020 Education Challenge and includes data from Task 3 and 4 (Wang et al. 2020). It contains responses from students who answered mathematics questions on Eedi, a global platform with millions of daily interactions.

| Dataset | # of interactions | # of students | # of questions | # of KCs |
|---|---|---|---|---|
| **AL2005** | 607,021 | 574 | 173,113 | 112 |
| **BD2006** | 1,817,458 | 1,145 | 129,263 | 493 |
| **NIPS34** | 1,382,678 | 4,918 | 948 | 57 |

Table 2: Data statistics of three widely used KT datasets.

## Baselines

We compare our model with the following baselines: (1) forgetting KT models: DKT-Forget (Nagatani et al. 2019), HawkesKT (Wang et al. 2021), and LPKT (Shen et al. 2021); (2) attention based KT models: SAKT (Pandey and Karypis 2019), SAINT (Choi et al. 2020), simpleKT (Liu et al. 2023), sparseKT (Huang et al. 2023), and DTransformer (Yin et al. 2023); and (3) forgetting attention based KT models: AKT (Ghosh, Heffernan, and Lan 2020) and RKT (Pandey and Srivastava 2020); (4) other KT models: DKT (Piech et al. 2015), DKVMN (Zhang et al. 2017) and DIMKT (Shen et al. 2022). We also consider some variants of attention based KT models that do not include forgetting mechanisms (RKT-NF and AKT-NF).

## Experimental Settings

To fairly compare with state-of-the-art KT models, we perform standard 5-fold cross-validation for all combinations of models and datasets. We use early stopping when performance does not improve after 10 epochs. We utilize the Adam optimizer to train the models up to 200 epochs. We adopt the Bayesian search method to find the best hyperparameters.

The embedding dimension, the hidden state dimension, and the dimensions of the two prediction layers are set to values in the range [64, 256]. The learning rate, dropout rate, and random seed are set within the range [1e-3, 1e-4, 1e-5]; [0.05, 0.1, 0.3, 0.5]; and [42, 3407], respectively. Similar to existing works (Piech et al. 2015; Ghosh, Heffernan, and Lan 2020; Liu et al. 2022), we use the AUC and accuracy as evaluation metrics. Furthermore, when evaluating the length extrapolation of KT models, all models are trained on interaction sequences with the fixed length of 200 and evaluated on sequences with lengths at 200, 400, 600, 800 and 1000, respectively. All the experiments are conducted on GeForce RTX 4090 GPUs.

## Results

We provide experiments and analyses to demonstrate the effectiveness of the proposed LefoKT model. Insights, findings, and observations are summarized as follows.

**Observation 1. Modeling forgetting behavior greatly affects KT models. Attention based KT model with forgetting mechanisms, i.e. AKT, is still competitive.** Table 3 summarizes the AUC and accuracy performance results. From Table 3, we find the following results: (1) Modeling forgetting behavior improves AUC and accuracy performance of RKT and AKT. For example, RKT outperforms RKT-NF in AUC performance by 0.48%, 0.83% and 0.75% on the AL2005, BD2006 and NIPS34 datasets, respectively. Similarly, AKT outperforms AKT-NF by 1.27%, 0.89% and 0.13% on the AL2005, BD2006 and NIPS34 datasets, respectively. This is because RKT employs a kernel function with an exponentially decaying curve to model student forgetting behavior, while AKT integrates a novel monotonic attention mechanism. Although the methods of modeling student forgetting behavior vary, they all achieve performance improvements. (2) AKT outperforms nearly all other methods on the AL2005 and BD2006 datasets and remains competitive on the NIPS34 dataset. For example, AKT surpasses the second-best LPKT by 0.38% in AUC performance on the AL2005 dataset and outperforms the

| Model | AL2005 | | BD2006 | | NIPS34 | |
|---|---|---|---|---|---|---|
| | AUC | ACC | AUC | ACC | AUC | ACC |
| **DKT** | 0.8149±0.0011 | 0.8097±0.0005 | 0.8015±0.0008 | 0.8553±0.0002 | 0.7689±0.0002 | 0.7032±0.0004 |
| **DKT-Forget** | 0.8147±0.0013 | 0.8090±0.0005 | 0.7985±0.0013 | 0.8536±0.0004 | 0.7733±0.0003 | 0.7076±0.0002 |
| **HAWKES** | 0.8210±0.0012 | 0.8115±0.0009 | 0.8068±0.0010 | 0.8559±0.0005 | 0.7767±0.0010 | 0.7110±0.0007 |
| **DKVMN** | 0.8054±0.0011 | 0.8115±0.0009 | 0.7983±0.0009 | 0.8545±0.0002 | 0.7673±0.0004 | 0.7016±0.0005 |
| **LPKT** | 0.8268±0.0004 | 0.8154±0.0008 | 0.8056±0.0008 | 0.8547±0.0005 | 0.8004±0.0003 | 0.7309±0.0006 |
| **DIMKT** | 0.8277±0.0009 | 0.8109±0.0005 | 0.8167±0.0008 | 0.8579±0.0001 | 0.8030±0.0002 | 0.7312±0.0005 |
| **SAKT** | 0.7899±0.0036 | 0.7965±0.0019 | 0.7739±0.0015 | 0.8460±0.0004 | 0.7525±0.0009 | 0.6884±0.0009 |
| **SAINT** | 0.7715±0.0018 | 0.7755±0.0012 | 0.7791±0.0018 | 0.8445±0.0013 | 0.7895±0.0009 | 0.7204±0.0009 |
| **sparseKT** | 0.8120±0.0018 | 0.8000±0.0009 | 0.8094±0.0005 | 0.8563±0.0003 | 0.7997±0.0005 | 0.7287±0.0003 |
| **DTransformer** | 0.8188±0.0025 | 0.8043±0.0021 | 0.8093±0.0009 | 0.8555±0.0007 | 0.7994±0.0003 | 0.7295±0.0007 |
| **simpleKT** | 0.8254±0.0014 | 0.8067±0.0011 | 0.8151±0.0006 | 0.8567±0.0010 | 0.8035±0.0000 | 0.7328±0.0001 |
| **RKT-NF** | 0.7713±0.0005 | 0.7922±0.0017 | 0.7548±0.0043 | 0.8434±0.0007 | 0.7891±0.0004 | 0.7198±0.0004 |
| **RKT** | 0.7761±0.0010 | 0.7964±0.0021 | 0.7631±0.0013 | 0.8468±0.0005 | 0.7966±0.0011 | 0.7264±0.0010 |
| **AKT-NF** | 0.8179±0.0025 | 0.8037±0.0019 | 0.8119±0.0007 | 0.8573±0.0004 | 0.8020±0.0005 | 0.7312±0.0002 |
| **AKT** | 0.8306±0.0019 | 0.8124±0.0011 | 0.8208±0.0007 | 0.8587±0.0005 | 0.8033±0.0003 | 0.7323±0.0005 |

Table 3: The performance comparisons of KT models on three datasets.

| Model | AL2005 | | BD2006 | | NIPS34 | |
|---|---|---|---|---|---|---|
| | AUC | ACC | AUC | ACC | AUC | ACC |
| **LefoKT$_{\text{RKT}}$** | 0.7761±0.0010 | 0.7964±0.0021 | 0.7631±0.0013 | 0.8468±0.0005 | 0.7966±0.0011 | 0.7264±0.0010 |
| **LefoKT$_{\text{RKT-NF}}$+ALiBi** | 0.7716±0.0008 | 0.7914±0.0007 | 0.7567±0.0045 | 0.8416±0.0046 | 0.7883±0.0006 | 0.7188±0.0007 |
| **LefoKT$_{\text{RKT-NF}}$+KERPLE-Log** | 0.7856±0.0009 | 0.8010±0.0008 | 0.7736±0.0019 | 0.8467±0.0005 | 0.7999±0.0002 | 0.7296±0.0002 |
| **LefoKT$_{\text{RKT-NF}}$+KERPLE-Power** | **0.7857±0.0005** | **0.8013±0.0008** | **0.7740±0.0019** | 0.8466±0.0004 | **0.8003±0.0004** | **0.7297±0.0009** |
| **LefoKT$_{\text{RKT-NF}}$+T5** | 0.7753±0.0008 | 0.7962±0.0005 | 0.7587±0.0026 | 0.8428±0.0025 | 0.7965±0.0005 | 0.7266±0.0006 |
| **LefoKT$_{\text{RKT-NF}}$+FIRE** | 0.7835±0.0011 | 0.8005±0.0009 | 0.7739±0.0010 | **0.8475±0.0005** | 0.8000±0.0001 | 0.7295±0.0007 |
| **LefoKT$_{\text{RKT-NF}}$+SandWich** | 0.7829±0.0014 | 0.8000±0.0015 | 0.7710±0.0010 | 0.8460±0.0012 | 0.7985±0.0004 | 0.7284±0.0004 |
| **LefoKT$_{\text{AKT}}$** | 0.8306±0.0019 | **0.8124±0.0011** | 0.8208±0.0007 | 0.8587±0.0005 | 0.8033±0.0003 | 0.7323±0.0005 |
| **LefoKT$_{\text{AKT-NF}}$+ALiBi** | **0.8317±0.0021** | 0.8110±0.0009 | **0.8247±0.0006** | **0.8605±0.0012** | **0.8045±0.0003** | **0.7340±0.0004** |
| **LefoKT$_{\text{AKT-NF}}$+KERPLE-Log** | 0.8276±0.0012 | 0.8115±0.0015 | 0.8230±0.0016 | 0.8603±0.0010 | 0.8040±0.0010 | 0.7333±0.0009 |
| **LefoKT$_{\text{AKT-NF}}$+KERPLE-Power** | 0.8258±0.0022 | 0.8094±0.0018 | 0.8207±0.0013 | 0.8588±0.0011 | 0.8041±0.0011 | 0.7338±0.0009 |
| **LefoKT$_{\text{AKT-NF}}$+T5** | 0.8260±0.0029 | 0.8073±0.0019 | 0.8201±0.0017 | 0.8589±0.0009 | 0.8035±0.0003 | 0.7326±0.0005 |
| **LefoKT$_{\text{AKT-NF}}$+FIRE** | 0.8221±0.0008 | 0.8097±0.0009 | 0.8116±0.0007 | 0.8559±0.0005 | 0.8027±0.0005 | 0.7319±0.0007 |
| **LefoKT$_{\text{AKT-NF}}$+SandWich** | 0.8275±0.0024 | 0.8100±0.0020 | 0.8234±0.0019 | 0.8600±0.0012 | 0.8037±0.0009 | 0.7334±0.0008 |

Table 4: The performance of attention based KT models with different RPEs.

second-best SimpleKT by 0.57% in AUC performance on the BD2006 dataset. In this paper, we mainly focus on how to better model student forgetting behavior to improve the performance of attention based KT models.

While the increases in AUC and accuracy shown in Table 3 are less than 1% compared to the best baseline across the three datasets, this improvement is noteworthy. Recent benchmark studies indicate that many reported performance gains are questionable due to flawed evaluation practices (Liu et al. 2022). Their study showed that since 2015, there was only a 3.5% improvement in overall KT prediction performance. Our study strictly adheres to their evaluation protocol and involves an exhaustive hyperparameter search for each baseline to ensure fair and reliable comparisons.

**Observation 2. RPEs significantly improve attention based KT models performance.** To thoroughly explore the impact of RPEs on attention based KT models, we conduct experiments using the LefoKT framework. We select RKT, RKT-NF, AKT, and AKT-NF as special cases of LefoKT, naming them LefoKT$_{\text{RKT}}$, LefoKT$_{\text{RKT-NF}}$, LefoKT$_{\text{AKT}}$, and LefoKT$_{\text{AKT-NF}}$, respectively.

Table 4 summarizes the performance results. From Table 4, we find that incorporating different RPEs into attention based KT models without the corresponding forgetting module can significantly improve performance compared to attention based KT models with the forgetting module. For example, LefoKT$_{\text{RKT-NF}}$ with KERPLE-Log, KERPLE-Power, FIRE, and SandWich each outperforms LefoKT$_{\text{RKT}}$ (with a forgetting module) in AUC performance on the AL2005 dataset. Similarly, LefoKT$_{\text{AKT-NF}}$ with ALiBi, KERPLE-Log, and SandWich each outperforms LefoKT$_{\text{AKT}}$ (with a forgetting module) in AUC performance on the BD2006 dataset. Among them, LefoKT$_{\text{RKT-NF}}$ with KERPLE-Power and LefoKT$_{\text{AKT-NF}}$ with ALiBi perform best. The AUC performance of LefoKT$_{\text{RKT-NF}}$ with KERPLE-Power is 0.96% higher than LefoKT$_{\text{RKT}}$, while the AUC performance of LefoKT$_{\text{AKT-NF}}$ with ALiBi is 0.11% higher than LefoKT$_{\text{AKT}}$.

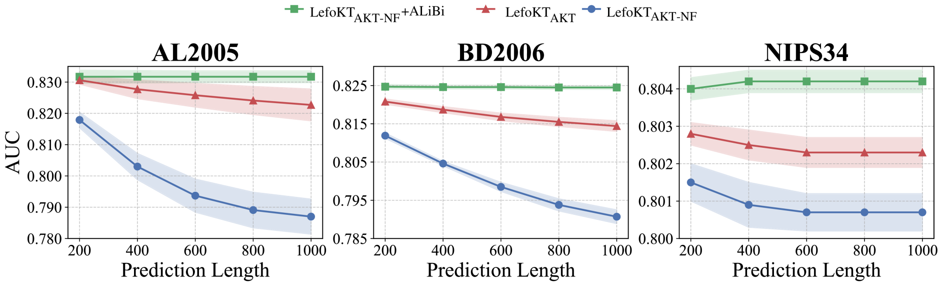


Figure 4: Extrapolation performance comparison for various prediction lengths across three datasets.

We hypothesize that this might be because RPEs can, to some extent, also model forgetting behavior. Subsequent experiments confirmed our hypothesis.

**Observation 3. RPEs essentially model student forgetting behavior and are more effective than other methods of modeling forgetting behavior.** Both KERPLE-Power and ALiBi inject positional information into attention scores by adding temporal biases. To intuitively understand what information is being injected, we visualize the values of the last row of the bias matrices for KERPLE-Power and ALiBi. As shown in Figure 5, the different colored curves represent different attention heads. We can find that the normalized decay rates differ across different heads. For KERPLE-Power, there is an exponential decay pattern, whereas for ALiBi, there is a linear decay pattern. According to psychological research (Wixted and Ebbesen 1991; Fisher and Radvansky 2022), students exhibit the different forgetting behavior and rate in various learning contexts. When the learning content is complex, forgetting follows an exponential pattern, while when the content is simpler, it follows a linear pattern. RPEs essentially model the various forgetting behavior.

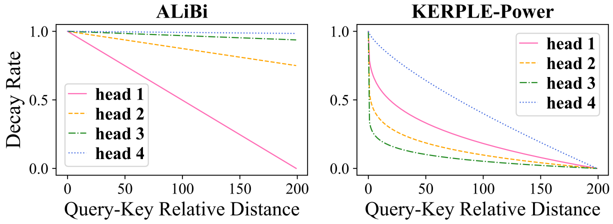


Figure 5: Comparison of different forgetting decay patterns.

To further verify our hypothesis that RPEs effectively model forgetting behavior, we incorporated RPEs into LefoKT$_{\text{RKT}}$ and LefoKT$_{\text{AKT}}$, both of which include exponential forgetting modules. We theorized that if RPEs indeed modeled forgetting, combining them with dedicated forgetting modules would lead to attention entropy redundancy or interference. The results in Table 5 support this hypothesis: adding RPEs to models with forgetting modules decreased performance. This suggests that RPEs and existing forgetting modules likely capture similar information, resulting in overfitting or feature redundancy. The performance decline further confirms that RPEs effectively model student forgetting, more efficiently than existing methods.

| Model | AL2005 | BD2006 | NIPS34 |
|---|---|---|---|
| **LefoKT$_{\text{RKT}}$** | 0.7761±0.0010 | 0.7631±0.0013 | 0.7966±0.0011 |
| **LefoKT$_{\text{RKT-NF}}$+K-P** | **0.7857±0.0005** | **0.7740±0.0019** | **0.8003±0.0004** |
| **LefoKT$_{\text{RKT}}$+K-P** | 0.7837±0.0008 | 0.7735±0.0014 | 0.8000±0.0002 |
| **LefoKT$_{\text{AKT}}$** | 0.8306±0.0019 | 0.8208±0.0007 | 0.8033±0.0003 |
| **LefoKT$_{\text{AKT-NF}}$+ALiBi** | **0.8317±0.0021** | **0.8247±0.0006** | **0.8045±0.0003** |
| **LefoKT$_{\text{AKT}}$+ALiBi** | 0.8310±0.0010 | 0.8199±0.0008 | 0.8032±0.0002 |

Table 5: Ablation experiments of forgetting module. K-P is KERPLE-Power.

**Observation 4. The proposed LefoKT model also demonstrates an excellent capability of length extrapolation.** The ability of length extrapolation enables attention based KT models to train on short interaction sequences and continue to perform well across longer sequences at the prediction stage, which is a significant challenge to attention based KT models (Press, Smith, and Lewis 2022; Li et al. 2024b). To evaluate this ability, we train LefoKT$_{\text{AKT}}$, LefoKT$_{\text{AKT-NF}}$ and LefoKT$_{\text{AKT-NF}}$+ALiBi on student interaction sequences of length 200 and evaluate them on sequences of length 200, 400, 600, 800 and 1000, respectively. As shown in Figure 4, we observe that LefoKT model maintains stable AUC performance with longer sequences, while LefoKT$_{\text{AKT-NF}}$ and LefoKT$_{\text{AKT}}$ exhibit varying degrees of performance drop. This indicates that our LefoKT model ( LefoKT$_{\text{AKT-NF}}$+ALiBi) exhibits strong ability of length extrapolation. This length extrapolation ability of LefoKT can be attributed to the introduction of RPEs. As sequence length increases, RPEs enable attention scores to attenuate, effectively modeling the forgetting process by focusing on nearby interactions and gradually discounting distant ones.

## Conclusion

In this paper, we introduce LefoKT, a flexible and forgetting-aware framework that uses relative forgetting attention to capture chronological order and model time-dependent forgetting patterns. LefoKT also enables length extrapolation across varying sequence lengths, effectively capturing students' long-term forgetting behavior over ever-growing learning sequences. Experiments on three KT datasets demonstrate LefoKT's improved performance compared to baseline methods, particularly in modeling students' long-term forgetting processes.

## Acknowledgements

This work was supported in part by National Key R&D Program of China, under Grant No. 2022YFC3303600 and in part by Key Laboratory of Smart Education of Guangdong Higher Education Institutes, Jinan University (2022LSYS003).

## References

Abdelrahman, G.; Wang, Q.; and Nunes, B. 2023. Knowledge Tracing: A Survey. *ACM Computing Surveys*, 55(11): 224–261.

Bailey, C. D. 1989. Forgetting and the learning curve: A laboratory study. *Management Science*, 35(3): 340–352.

Chi, T.; Fan, T.; Ramadge, P. J.; and Rudnicky, A. 2022. KERPLE: Kernelized Relative Positional Embedding for Length Extrapolation. In *Proceedings of the 36th Conference on Neural Information Processing Systems*, 8386–8399.

Chi, T.-C.; Fan, T.-H.; Rudnicky, A.; and Ramadge, P. 2023. Dissecting Transformer Length Extrapolation via the Lens of Receptive Field Analysis. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, 13522–13537.

Choi, Y.; Lee, Y.; Cho, J.; Baek, J.; Kim, B.; Cha, Y.; Shin, D.; Bae, C.; and Heo, J. 2020. Towards an Appropriate Query, Key, and Value Computation for Knowledge Tracing. In *Proceedings of the 7th ACM Conference on Learning at Scale*, 341–344.

Corbett, A. T.; and Anderson, J. R. 1994. Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User-adapted Interaction*, 10(4): 253–278.

Fisher, J. S.; and Radvansky, G. A. 2022. Degree of learning and linear forgetting. *Quarterly Journal of Experimental Psychology*, 75(8): 1483–1496.

Ghosh, A.; Heffernan, N.; and Lan, A. S. 2020. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2330–2339.

Huang, S.; Liu, Z.; Zhao, X.; Luo, W.; and Weng, J. 2023. Towards Robust Knowledge Tracing Models via k-Sparse Attention. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2441–2445.

Kazemnejad, A.; Padhi, I.; Ramamurthy, K. N.; Das, P.; and Reddy, S. 2023. The Impact of Positional Encoding on Length Generalization in Transformers. In *Proceedings of the 37th Conference on Neural Information Processing Systems*, 24892–24928.

Li, S.; You, C.; Guruganesh, G.; Ainslie, J.; Ontanon, S.; Zaheer, M.; Sanghai, S.; Yang, Y.; Kumar, S.; and Bhojanapalli, S. 2024a. Functional Interpolation for Relative Positions improves Long Context Transformers. In *Proceedings of the 12th International Conference on Learning Representations*.

Li, X.; Bai, Y.; Guo, T.; Liu, Z.; Huang, Y.; Zhao, X.; Xia, F.; Luo, W.; and Weng, J. 2024b. Enhancing Length Generalization for Attention Based Knowledge Tracing Models with Linear Biases. In *Proceedings of the 32th International Joint Conference on Artificial Intelligence*, 5918–5926.

Li, X.; Bai, Y.; Guo, T.; Zheng, Y.; Hou, M.; Zhan, B.; Huang, Y.; Liu, Z.; Gao, B.; and Luo, W. 2024c. Extending Context Window of Attention Based Knowledge Tracing Models via Length Extrapolation. In *Proceedings of the 27th European Conference on Artificial Intelligence*, 1479–1486.

Li, Z.; Yang, J.; Wang, J.; Shi, L.; Feng, J.; and Stein, S. 2024d. LBKT: A LSTM BERT-based Knowledge Tracing Model for Long-sequence Data. In *International Conference on Intelligent Tutoring Systems*, 174–184.

Liu, Z.; Liu, Q.; Chen, J.; Huang, S.; and Luo, W. 2023. simpleKT: A Simple But Tough-to-Beat Baseline for Knowledge Tracing. In *Proceedings of the 11th International Conference on Learning Representations*.

Liu, Z.; Liu, Q.; Chen, J.; Huang, S.; Tang, J.; and Luo, W. 2022. pyKT: A Python Library to Benchmark Deep Learning based Knowledge Tracing Models. In *Proceedings of 36th Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 18542–18555.

Luo, R.; Huang, H.; Yu, S.; Zhang, X.; and Xia, F. 2024. FairGT: A Fairness-aware Graph Transformer. In *Proceedings of the 32th International Joint Conference on Artificial Intelligence*, 449–457.

Nagatani, K.; Zhang, Q.; Sato, M.; Chen, Y.; Chen, F.; and Ohkuma, T. 2019. Augmenting Knowledge Tracing by Considering Forgetting Behavior. In *Proceedings of the ACM Web Conference*, 3101–3107.

Pandey, S.; and Karypis, G. 2019. A Self-Attentive Model for Knowledge Tracing. In *Proceedings of the 12th International Conference on Educational Data Mining*, 384–389.

Pandey, S.; and Srivastava, J. 2020. RKT: Relation-Aware Self-Attention for Knowledge Tracing. In *The 29th ACM International Conference on Information and Knowledge Management*, 1205–1214.

Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L. J.; and Sohl-Dickstein, J. 2015. Deep Knowledge Tracing. In *Proceedings of the 28th Conference on Neural Information Processing Systems*, 505–513.

Press, O.; Smith, N.; and Lewis, M. 2022. Train Short, Test Long: Attention with Linear Biases Enables Input Length Extrapolation. In *Proceedings of the 10th International Conference on Learning Representations*.

Qin, Z.; Zhong, Y.; and Deng, H. 2024. Exploring Transformer Extrapolation. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, volume 38, 18897–18905.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *The Journal of Machine Learning Research*, 21(140): 1–67.

Shehata, S.; Calonge, D. S.; Purnell, P.; and Thompson, M. 2023. Enhancing video-based learning using knowledge tracing: Personalizing students' learning experience with ORBITS. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications*, 100–107.

Shen, S.; Huang, Z.; Liu, Q.; Su, Y.; Wang, S.; and Chen, E. 2022. Assessing student's dynamic knowledge state by exploring the question difficulty effect. In *Proceedings of the 45th international ACM SIGIR Conference on Research and Development in Information Retrieval*, 427–437.

Shen, S.; Liu, Q.; Chen, E.; Huang, Z.; Huang, W.; Yin, Y.; Su, Y.; and Wang, S. 2021. Learning Process-consistent Knowledge Tracing. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 1452–1460.

Stamper, J.; and Pardos, Z. A. 2016. The 2010 KDD Cup Competition Dataset: Engaging the machine learning community in predictive learning analytics. *Journal of Learning Analytics*, 3(2): 312–316.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 6000–6010.

Wang, C.; Ma, W.; Zhang, M.; Lv, C.; Wan, F.; Lin, H.; Tang, T.; Liu, Y.; and Ma, S. 2021. Temporal cross-effects in knowledge tracing. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, 517–525.

Wang, Z.; Lamb, A.; Saveliev, E.; Cameron, P.; Zaykov, Y.; Hernández-Lobato, J. M.; Turner, R. E.; Baraniuk, R. G.; Barton, C.; Jones, S. P.; et al. 2020. Instructions and Guide for Diagnostic Questions: The NeurIPS 2020 Education Challenge. *arXiv preprint arXiv:2007.12061*.

Wixted, J. T.; and Ebbesen, E. B. 1991. On the form of forgetting. *Psychological Science*, 2(6): 409–415.

Yin, Y.; Dai, L.; Huang, Z.; Shen, S.; Wang, F.; Liu, Q.; Chen, E.; and Li, X. 2023. Tracing Knowledge Instead of Patterns: Stable Knowledge Tracing with Diagnostic Transformer. In *Proceedings of the ACM Web Conference*, 855–864.

Zhang, J.; Shi, X.; King, I.; and Yeung, D.-Y. 2017. Dynamic Key-value Memory Networks for Knowledge Tracing. In *Proceedings of the ACM Web Conference*, 765–774.