Enhancing the Knowledge Tracing via a Plug-In Guided Diffusion Model

Shuaishuai Zu[1], Jihao Zhao[1], Biao Qin[1]*

[1]School of Information, Renmin University of China, Beijing, China
{zushuaishuai, zhaojihao, qinbiao}@ruc.edu.cn

Abstract

Knowledge tracing (KT) refers to the problem of predicting students' future performance given their past performance. Scrutinizing previous studies, we can summarize a common *learn-to-predict* paradigm: a KT model first learns the student's latent knowledge states from historical question-solving learning interactions and then directly predicts whether the student could correctly answer new questions. Alongside the paradigm, existing KT models are dedicated to tailoring refinements for improving predictive performance. However, this has led to increasing model complexity and reduced usability. Inspired by the diagnosis process of human teachers, they conduct correctness prediction based on the students' responses, which are further derived from their latent knowledge states. To achieve this, we propose a novel plug-in Guided diffusiOn mODule (GOOD), which reframes the KT problem as a *learn-generate-to-predict* paradigm. Specifically, we first employ an existing KT backbone to learn the student's evolving latent knowledge states, subsequently feeding these into our GOOD. Next, GOOD employs a person-wise noise scheduling strategy to add noise to the target responses in the diffusion process, thereby exploring the underlying distribution of response space. Then, GOOD designs a flexible transformer-modulated denoising network to generate target responses utilizing the latent knowledge states as conditional guidance in the reverse process. Finally, the generated responses can explicitly reflect the student's performance, thereby facilitating the correctness prediction. Extensive experiments on four datasets have verified the effectiveness of GOOD in boosting existing KT models to achieve state-of-the-art performance, as well as its generalizability as a flexible plugin.

Introduction

Knowledge tracing (Abdelrahman, Wang, and Nunes 2023) plays a critical role in multimedia intelligent tutoring systems and adaptive learning platforms. It aims to predict learners' future performance (whether they could respond to assessment questions correctly) based on their past performance, as illustrated in Figure 1(a). Accurate KT enables personalized learning experiences, helping educators identify knowledge gaps and recommend appropriate learning resources to students (Hwang and Lee 2025).

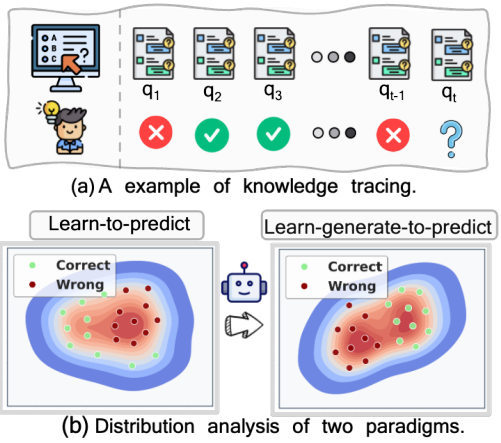

Figure 1: Comparison of different KT paradigms.

Currently, the field of KT has garnered substantial research interest. Scrutinizing recent research on the KT task, we summarize that most studies follow a *learn-to-predict* paradigm (Shen et al. 2024): a KT model first learns the student's latent knowledge states from historical question-solving interactions and then utilizes these to directly predict whether the student could correctly answer a new question. Within the paradigm, recent efforts can be grouped into two primary directions. The first focuses on incorporating various types of side information to enrich the representation ability of learning interactions, thereby better capturing student-question relations. They design specialized model structures to leverage structural relationships like question-concept graphs (Wang et al. 2025), extract semantic information of question text (Liu et al. 2019), or consider temporal dynamics (Wang et al. 2021), etc. The second direction focuses on integrating advanced deep learning techniques to capture the intricate dependencies inherent in learning trajectories. These studies are mainly conducted on adapting sophisticated neural network architectures (e.g., recurrent neural networks (Nagatani et al. 2019), transformers (Ghosh, Heffernan, and Lan 2020; Zhou et al. 2025; Huang et al. 2023; Wang et al. 2023; Fu et al. 2024), and diffusion model (Kuo et al. 2024)) and incorporating specialized frameworks (e.g., contrastive learning (Yin et al. 2023),

*Corresponding author.

multi-objective optimization (Zu et al. 2024)). Although effective, these methods generally depend on tailored refinements, making it challenging for subsequent research to derive substantial insights from them. Looking back to the diagnosis process of human teachers, they generally characterize students' performance based on their actual responses. It prompts an intuitive question: Can we go one step further to generate students' target responses from their latent knowledge states, thereby facilitating the predictive performance for existing KT methods?

In this paper, we propose a novel plug-in **G**uided diffusi**O**n m**OD**ule, termed **GOOD**, which solves the KT problem through a *learn-generate-to-predict* paradigm. GOOD explicitly models the process of generating students' target responses and can be seamlessly integrated into existing KT models (e.g., AKT (Ghosh, Heffernan, and Lan 2020), FoLiBiKT (Wang et al. 2023), DisKT (Zhou et al. 2025)) as a plug-in module. We first utilize a KT backbone to learn the student's evolving latent knowledge states and subsequently input them into GOOD. Then, to better adapt to the KT task, we propose two key components within our GOOD. In the diffusion process, unlike the conventional token-wise noise addition strategy (Ho, Jain, and Abbeel 2020), we employ a person-wise noise scheduling strategy on the representations of the target responses. This strategy adds the same-level noises across student's entire learning interactions rather than adding varying noises at each interaction (Qi et al. 2024), ensuring consistency in the generative process that aligns with the coherent nature of whole learning trajectories. In the reverse process, we design a transformer-modulated denoising network (TMDN) that harnesses the power of transformers to extract insights from past latent knowledge states, facilitating accurate distribution estimation for target responses. Inside TMDN, we devise a knowledge-aware adaptive layer normalization mechanism that aggregates latent knowledge states locally to yield robust conditional guidance. Finally, predictions are made by considering both latent knowledge states and the generated responses. Figure 1(b) illustrates t-SNE (Maaten and Hinton 2008) visualizations of input representations to the final prediction head of DKT under two paradigms, where each point represents the decision-making basis for each question. Incorporating GOOD into DKT results in clearer boundaries, facilitating subsequent correctness predictions. The contributions are as follows:

- We reshape the KT task as a new *learn-generate-to-predict* paradigm via a novel plug-in GOOD, which can be seamlessly integrated into existing KT models for boosting predictive performance.
- Within GOOD, we propose the following: (1) a person-wise noise scheduling strategy to maintain sequence consistency in the diffusion process; (2) a TMDN with knowledge-aware adaptive layer normalization to accurately guide response generation in the reverse process.
- Extensive experiments on four datasets demonstrate that GOOD enables prevailing KT models to achieve significant and consistent performance improvements, validating its effectiveness and generalizability.

Related Work

Knowledge Tracing has emerged as a fundamental task in educational data mining, aiming to monitor students' learning processes and predict their future performance (Abdelrahman, Wang, and Nunes 2023). Early approaches were mainly probabilistic, such as Bayesian Knowledge Tracing (BKT) (Yudelson, Koedinger, and Gordon 2013), which treats students' knowledge as binary latent variables and uses Hidden Markov Models (Eddy 1996) to estimate conceptual mastery over time. With the rise of deep learning, recent studies can be broadly grouped into two directions. The first direction focuses on enriching interaction representations with auxiliary information. Graph-based approaches like DGEKT (Cui et al. 2024) and DyKT (Cheng et al. 2024) leverage structural relationships between concepts and questions. EKT (Liu et al. 2019) incorporates question text and difficulty to integrate semantic content. Temporal dynamics are modeled by DKT-Forgetting (Nagatani et al. 2019) and HawkesKT (Wang et al. 2021), which consider temporal intensity patterns. The second direction centers on advanced deep learning techniques for capturing intricate learning patterns. Deep Knowledge Tracing (DKT) (Piech et al. 2015) introduces recurrent neural networks to model complex temporal dependencies. Memory-enhanced models such as DKVMN (Zhang et al. 2017) and SKVMN (Abdelrahman and Wang 2019) add external memory mechanisms for flexible knowledge tracking. Attention-based approaches, including SAKT (Pandey and Karypis 2019), AKT (Ghosh, Heffernan, and Lan 2020), SAINT (Choi et al. 2020), and DisKT (Zhou et al. 2025), apply self-attention and context-aware mechanisms to model long-range dependencies. Furthermore, CL4KT (Lee et al. 2022) and Dtransformer (Yin et al. 2023) introduce contrastive learning frameworks, while QDKT (Liu, Zhan, and Kim 2024) and SimpleKT (Liu et al. 2023) design auxiliary-task modules. Despite these advances, most models still follow a *learn-to-predict* paradigm, directly mapping from learned knowledge states to predictions. This may oversimplify students' problem-solving, where they typically first generate responses based on their understanding, making it easier to predict performance from generated responses than from latent states alone.

Diffusion Model

Diffusion models have emerged as a powerful class of generative models that learn to reverse a gradual noising process to generate high-quality samples (Yang et al. 2023). With significant advances achieved by Denoising Diffusion Probabilistic Models (DDPM) (Ho, Jain, and Abbeel 2020) and Denoising Diffusion Implicit Models (DDIM) (Song, Meng, and Ermon 2020). Diffusion models have demonstrated remarkable success in image generation (Yang et al. 2024), achieving state-of-the-art (SOTA) results in terms of both sample quality and diversity (Cao et al. 2024). The core principle of diffusion models involves a forward process that gradually adds Gaussian noise to data samples until they become pure noise, and a reverse process that learns to denoise samples step by step to recover the original data distribution. During training, the model learns to predict the noise added

at each timestep, while during inference, it iteratively denoises random noise to generate new samples. The success of diffusion models stems from their stable training dynamics, high sample quality, and theoretical grounding in non-equilibrium thermodynamics (Hyvärinen and Dayan 2005). Recent developments have extended diffusion models to KT. For example, researchers utilize TabDDPM (Kotelnikov et al. 2023) to synthesize student interaction data (Kuo et al. 2024). MSKT employs a complex diffusion model to capture the dynamic transition of student knowledge states from resting to executing in latent space (Zhang, Ji, and Zhang 2024). Unlike them, we leverage the generative capability of diffusion models to extend the existing KT paradigm, working as a plugin.

Preliminary

Generally, diffusion models consist of two main processes: a diffusion process and a reverse process. The diffusion process defines a Markov chain that gradually adds Gaussian noise to the data $\mathbf{x}^0$ over T diffusion steps, eventually transforming it into pure Gaussian noise. At each step t, the forward diffusion process is defined as:

$$q(\mathbf{x}^t|\mathbf{x}^{t-1}) = \mathcal{N}(\mathbf{x}^t; \sqrt{1-\beta_t}\mathbf{x}^{t-1}, \beta_t\mathbf{I}), \tag{1}$$

where $\beta_t \in (0,1)$ is the variance schedule that controls the amount of noise added at timestep t. The noise schedule is typically chosen to be small and gradually increasing, such as a linear schedule: $\beta_t = \beta_1 + \frac{t-1}{T-1}(\beta_t - \beta_1)$. By the reparameterization trick, the forward process can be written as:

$$\mathbf{x}^t = \sqrt{1-\beta_t}\mathbf{x}^{t-1} + \sqrt{\beta_t}\boldsymbol{\epsilon}_{t-1}, \tag{2}$$

where $\boldsymbol{\epsilon}_{t-1} \sim \mathcal{N}(0, \mathbf{I})$ is standard Gaussian noise. A key property of the forward process is that we can sample $\mathbf{x}^t$ directly from $\mathbf{x}^0$ without iterating through all intermediate steps. Let $\alpha^t = 1 - \beta_t$ and $\bar{\alpha}_t = \prod_{s=1}^{t} \alpha_s$, then:

$$q(\mathbf{x}^t|\mathbf{x}^0) = \mathcal{N}(\mathbf{x}^t; \sqrt{\bar{\alpha}_t}\mathbf{x}^0, (1-\bar{\alpha}_t)\mathbf{I}). \tag{3}$$

This can be expressed as:

$$\mathbf{x}^t = \sqrt{\bar{\alpha}_t}\mathbf{x}^0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \tag{4}$$

where $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. As $t \to T$, we have $\bar{\alpha}_t \to 0$, ensuring that $\mathbf{x}^t$ approaches pure Gaussian noise.

The reverse process aims to recover the original data by reversing the forward diffusion process step by step. The actual reverse process $q(\mathbf{x}^{t-1}|\mathbf{x}^t)$ is intractable, so we approximate it with a parameterized model $p_\theta(\mathbf{x}^{t-1}|\mathbf{x}^t)$:

$$p_\theta(\mathbf{x}^{t-1}|\mathbf{x}^t) = \mathcal{N}(\mathbf{x}^{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}^t, t), \Sigma_\theta(\mathbf{x}^t, t)), \tag{5}$$

where $\boldsymbol{\mu}_\theta$ and Σ_θ are neural networks that predict the mean and variance of the reverse distribution. The training objective for diffusion models is derived from the variational lower bound of the log-likelihood (Song et al. 2022). The loss function can be simplified to:

$$\mathcal{L} = \mathbf{E}_{t\sim\mathcal{U}(1,T),\mathbf{x}^0\sim q(\mathbf{x}^0),\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I})}\left[\|\boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\mathbf{x}^t, t)\|^2\right], \tag{6}$$

where $\boldsymbol{\epsilon}_\theta$ is the neural network that predicts the noise added at step t, and $\mathbf{x}^t = \sqrt{\bar{\alpha}_t}\mathbf{x}^0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$.

Method

Problem Definition

Let $\mathcal{S}_{1:n-1} = \{(q_i, r_i)\}_{i=1}^{n}$ denote a student's historical learning sequence upon timestamp n, where $c_n \in \mathcal{C}$ represents the concept id associated with the question q_i at timestamp i, and $r_i \in \{0, 1\}$ indicates the student's response correctness (1 for correct, 0 for incorrect). The KT task can be formalized as a supervised sequential prediction problem: given the student's past interactions $\mathcal{S}_{1:n-1}$, predict the probability of responding correctly to a future question q_n involving concept c_n.

$$P(r_n = 1|q_n, \mathcal{S}_{1:n-1}). \tag{7}$$

As illustrated in Figure 2, we follow a novel *learn-generate-to-predict* paradigm that consists of three stages: (1) learning student's evolving latent knowledge states from historical interactions using an existing KT backbone, (2) generating representations of target response through proposed GOOD conditioned on learned knowledge states and diffusion steps, and (3) predicting student performance by combining both the learned knowledge states and the generated response representations.

Stage 1: Learning Latent Knowledge States

We leverage existing KT models as backbones to learn students' latent knowledge states, including AKT (Ghosh, Heffernan, and Lan 2020), DisKT (Zhou et al. 2025), and FoLiBiKT (Wang et al. 2023), etc. Given the student's historical learning sequence $\{(q_i, r_i)\}_{i=1}^{n}$, the KT backbone first transforms the discrete interaction tuples into continuous embedding representations. The embedding process involves: $\mathbf{e}_{q_n} \in \mathbb{R}^{1\times d}$ denotes the representation of q_n and $\mathbf{e}_{r_n} \in \mathbb{R}^{1\times d}$ refers to the representation of correctness r_n. The interaction representation for timestamp i is formed by combining the question and correctness information:

$$\mathbf{x}_i = \mathbf{e}_{q_i} + \mathbf{e}_{r_i}. \tag{8}$$

The embedded interactions $\mathbf{E}_{1:n} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ are then processed through the KT backbone $f_{KT}(\cdot)$ to capture temporal dependencies and learning dynamics from learning behaviors:

$$\begin{aligned}&\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n \\ &\quad = f_{KT}(\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{n-1}; \mathbf{e}_{q_1}, \mathbf{e}_{q_2}, \cdots, \mathbf{e}_{q_n}),\end{aligned} \tag{9}$$

where $\mathbf{h}_t \in \mathbb{R}^{1\times d}$ encodes the student's accumulated knowledge for answering question q_n. These knowledge states serve as comprehensive representations of the student's cognitive state and will be fed into GOOD as generation guidance.

Stage 2: Guided Diffusion Module for Generation

Based on the learned knowledge states, GOOD models the response construction process that reflects students' cognitive understanding.

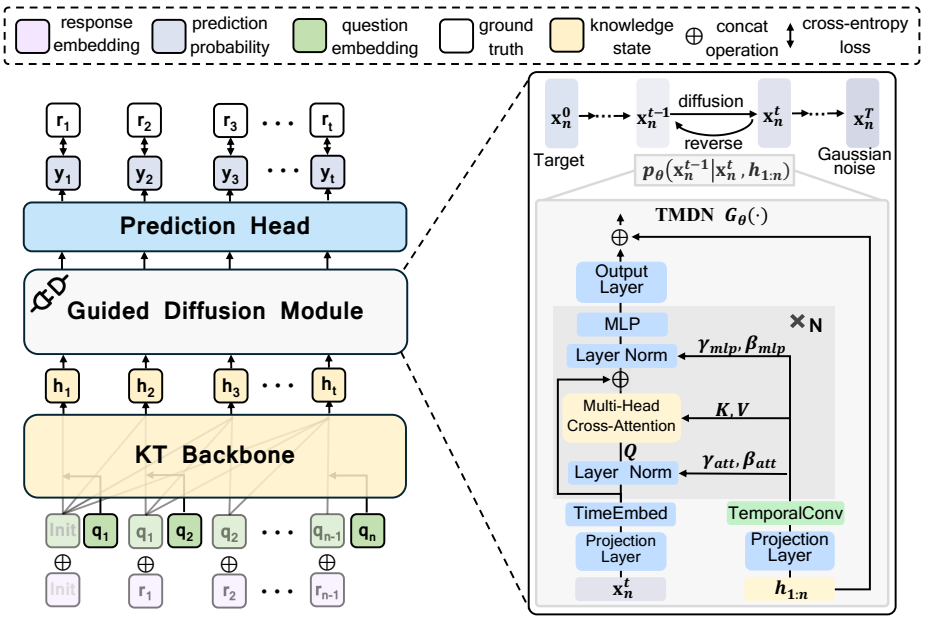

Figure 2: Overview of the *learn-generate-to-predict* paradigm implemented via our proposed plug-in **GOOD**.

Diffusion Process To better adapt the diffusion process to the KT task, we employ a person-wise noise scheduling strategy that departs from the conventional token-wise noise addition strategy. Let $\mathbf{x}_n^0$ denote the target response representation $\mathbf{x}_n$ that we aim to generate for answering q_n. The forward diffusion process gradually corrupts this representation by adding Gaussian noise over T steps:

$$q(\mathbf{x}_n^t|\mathbf{x}_n^{t-1}) = \mathcal{N}(\mathbf{x}_n^t; \sqrt{1-\beta_t}\mathbf{x}_n^{t-1}, \beta_t\mathbf{I}), \quad (10)$$

where $\beta_t \in (0,1)$ is the noise schedule at diffusion step t. Using the reparameterization property, we can directly sample the noisy response representation at diffusion step t:

$$\mathbf{x}_n^t = \sqrt{\bar{\alpha}_t}\mathbf{x}_n^0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad (11)$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I})$. Crucially, our person-wise noise scheduling strategy applies the same-level noises across $\mathbf{E}_{1:n}$ rather than varying noise at each interaction $\mathbf{x}_k \in \mathbf{E}_{1:n}$. This approach is motivated by the observation that student learning behaviors exhibit temporal consistency within a session (Shen et al. 2021).

Reverse Process In the reverse process, we design a TMDN that generates target response $\mathbf{x}_n^0$ by iteratively denoising the corrupted representations $\mathbf{x}_n^t$:

$$p_\theta(\mathbf{x}_n^{t-1}|\mathbf{x}_n^t, \mathbf{h}_{1:n}) = \mathcal{N}(\mathbf{x}_n^{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_n^t, \mathbf{h}_{1:n}, t), \Sigma_\theta(\mathbf{x}_n^t, \mathbf{h}_{1:n}, t)). \quad (12)$$

Unlike traditional diffusion models that predict noise, we can directly predict the target response representation $\mathbf{x}_n^0$ (Yang et al. 2023). Therefore, we can rewrite $\boldsymbol{\mu}_\theta$ as:

$$\boldsymbol{\mu}_\theta(\mathbf{x}_n^t, \mathbf{h}_{1:n}, t) = \sqrt{\bar{\alpha}_t}G_\theta(\mathbf{x}_n^t, \mathbf{h}_{1:n}, t) + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, \quad (13)$$

where $G_\theta(\cdot)$ represents our TMDN that exploits the power of transformers to extract essential insights from historical $\mathbf{h}_{1:n}$ as guidance. Our TMDN $G_\theta(\cdot)$ takes three inputs: the noisy response representation $\mathbf{x}_n^t$, the knowledge state conditioning $\mathbf{h}_{1:n}$, and the diffusion step t. These inputs are processed through dedicated projection layers containing Gaussian error linear unit (GELU) activation (Hendrycks and Gimpel 2016) followed by linear transformations:

$$\mathbf{x}_n^q = \text{Projection}(\mathbf{x}_n^t) + \text{TimeEmbed}(t), \quad (14)$$

$$\mathbf{x}_{1:n}^{kv} = \text{Projection}(\mathbf{h}_{1:n}). \quad (15)$$

We employs sinusoidal positional encoding operation $\text{TimeEmb}(t) \in \mathbb{R}^{1\times d}$ for the diffusion step t. To handle varying noise levels throughout the diffusion process, the TMDN incorporates a knowledge-aware adaptive layer normalization mechanism that generates adaptive parameters controlled by aggregated knowledge states:

$$\mathbf{x}_{1:n}^{kv} = \text{TemporalConv}(\mathbf{h}_{1:n}), \quad (16)$$

$$[\boldsymbol{\gamma}_{att}, \boldsymbol{\beta}_{att}] = \text{Linear}(\text{SiLU}(\mathbf{x}_n^{kv})), \quad (17)$$

$$[\boldsymbol{\gamma}_{mlp}, \boldsymbol{\beta}_{mlp}] = \text{Linear}(\text{SiLU}(\mathbf{x}_n^{kv})). \quad (18)$$

The $\text{TemporalConv}(\cdot)$ operation applies causal convolutions with residual connections to model local temporal patterns while implementing causal masking to prevent information leakage from future performance. The adaptive parameters modulate both attention and MLP layers via the adaptive layer normalization operation:

$$\text{AdaLN}(\mathbf{x}, \boldsymbol{\gamma}, \boldsymbol{\beta}) = \boldsymbol{\gamma} \odot \text{LayerNorm}(\mathbf{x}) + \boldsymbol{\beta}. \quad (19)$$

Next, the normalized query representations $\mathbf{x}_n^q$ participate in multi-head cross-attention layer. This enables dynamic adaptation based on both noise corruption level and aggregated knowledge context, allowing the model to adjust its

denoising strategy throughout the reverse process.

$$\alpha_{i,j} = \text{softmax}\left(\frac{Q_n K_n^T}{\sqrt{d}} \cdot M_{i,j}\right), \tag{20}$$

$$\mathbf{x}_n^{att} = \sum_{k=1}^{k=n} \alpha_{n,k} \cdot V_j, \tag{21}$$

where $Q_n = \mathbf{x}_n^q \mathbf{W}^q$, $K_n = \mathbf{x}_n^{kv} \mathbf{W}^k$, $V_n = \mathbf{x}_n^{kv} \mathbf{W}^v$ and $M_{i,j} = \mathbf{1}_{i \le j}$ enforce causal masking to prevent information leakage.

$$\mathbf{x}_n^{out} = \mathbf{x}_n^{att} + \text{MLP}(\text{AdaLN}(\mathbf{x}_{att}, \boldsymbol{\gamma}_{mlp}, \boldsymbol{\beta}_{mlp})), \tag{22}$$

where $\text{MLP}(\cdot)$ consists of two linear layers with GELU activation and dropout for regularization. The model then applies the second set of AdaLN parameters to modulate MLP processing. Then, the refined representations are transmitted into the output layer that shares the same structure as the projection layer:

$$\hat{\mathbf{x}}_n^0 = \text{OutputLayer}(\mathbf{x}_n^{out}). \tag{23}$$

Stage 3: Performance Prediction

Finally, predictions are made by considering both the $\mathbf{h}_n$ learned from the KT backbone and the $\hat{\mathbf{x}}_n^0$ generated from our GOOD.

$$y_n = \text{Sigmoid}(\text{MLP}([\mathbf{h}_n \oplus \mathbf{x}_n^0])). \tag{24}$$

This integration results in more discriminative prediction-making boundaries and improved predictive performance compared to the previous paradigm.

Training Objective

Combined with GOOD, the enhanced model is trained with a combined objective that includes both performance prediction and response generation tasks. For the performance prediction task, we use a binary cross-entropy loss (Zhang and Sabuncu 2018):

$$\mathcal{L}_{\text{KT}} = \sum_{n=1}^{N} -r_n \log y_n - (1 - r_n) \log(1 - y_n), \tag{25}$$

where r_i denotes the real response for question q_i. For the response generation task, the model is trained using direct reconstruction loss rather than noise prediction loss (Yang et al. 2023):

$$\mathcal{L}_{\text{GEN}} = \mathbf{E}_{t \sim \mathcal{U}(1,T), \mathbf{x}_n^0, \boldsymbol{\epsilon}} \left[\|\mathbf{x}_n^0 - G_\theta(\mathbf{x}_n^t, \mathbf{h}_{1:n}, t)\|^2 \right]. \tag{26}$$

The overall training objective combines both losses with a balanced hyperparameter λ:

$$\mathcal{L} = \mathcal{L}_{\text{KT}} + \lambda \mathcal{L}_{\text{GEN}}. \tag{27}$$

Experiments

We conduct extensive experiments to answer the following four questions to evaluate the effectiveness of our GOOD:

- **RQ1:** How effective is GOOD in improving the predictive performance of existing KT models?
- **RQ2:** How effective are the key components of GOOD?
- **RQ3:** What are the impacts of hyperparameter λ?
- **RQ4:** How does GOOD perform in terms of boosting prediction quality and computational efficiency?

Dataset	Students	Exercises	Concepts	Interactions
Assist09	4,151	26,688	123	325,637
Assist12	22,422	45,543	99	1,839,429
Ednet	50,000	12,117	189	676,276
Spanish	182	409	221	578,726

Table 1: Statistics of experimental datasets

Datasets and Baselines

We conduct experiments on four widely-used benchmark datasets[1]: Assist09, Assist12, Ednet, and Spanish. Following the data pre-processing approach in (Zhou et al. 2025), the statistical information after processing is shown in Table 1. We integrate our GOOD into five selected representative KT models: DKT, LPKT, AKT, DisKT, and FoLiBiKT, and also apply HD-KT (Ma et al. 2024) on them to validate the effectiveness and universality of our method. Note that HD-KT is a SOTA method, which enhances existing KT models with a plug-in **H**ybrid learning interaction **D**enoising approach (**HD**). For fair comparison, we set N to 4 in GOOD's TMDN $G_\theta(\cdot)$, ensuring that the parameter count of GOOD is close to but less than that of HD. Moreover, we also include three strong KT baselines: DTransformer (Yin et al. 2023), SparseKT (Huang et al. 2023), and SimpleKT (Liu et al. 2023).

Experimental Settings

All experiments are conducted on a Linux server with an NVIDIA 4090D (48G). We perform 5-fold cross-validation and report average results. In each fold, data are split into 70% for training, 10% for validation, and 20% for testing. All models use the AdamW optimizer (Loshchilov, Hutter et al. 2017) with a learning rate of 1e-3, weight decay of 1e-5, and gradient clipping at norm 1.0. Batch size is set to 256 for smaller datasets (Assist09, Spanish) and 512 for larger datasets (Ednet, Assist12). We apply dropout with a rate of 0.1 throughout the network to prevent overfitting. During training, we employ $T = 1000$ diffusion steps (Ho, Jain, and Abbeel 2020) to ensure noise distribution learning and stable convergence. For inference, we adopt the DDIM (Song, Meng, and Ermon 2020) sampling strategy with 50 steps, which provides significant efficiency advantages. For reproducibility, we set the same random seed for all experiments.

Comparison studies (RQ1)

Table 2 summarizes experimental results across four benchmark datasets, revealing the consistent superiority of GOOD-enhanced KT models. First, GOOD enhances all five selected KT models via plug-and-play integration, yielding substantial performance gains. For example, FoLiBiKT-GOOD achieves average improvements of 2.3% AUC, 1.6% ACC, and 1.2% RMSE reduction compared to vanilla FoLiBiKT. Second, compared with five HD-enhanced KT models, GOOD-enhanced counterparts consistently achieve superior performance. Notably, AKT-GOOD achieves 0.7844

[1]All datasets can be found at https://base.ustc.edu.cn/data

Datasets	Assist09			Assist12			Ednet			Spanish		
Metrics	AUC↑	ACC↑	RMSE↓	AUC↑	ACC↑	RMSE↓	AUC↑	ACC↑	RMSE↓	AUC↑	ACC↑	RMSE↓
DTransformer	0.7508	0.7042	0.4505	0.7551	0.7433	0.4208	0.6978	0.6559	0.4799	0.8170	0.7513	0.4108
SparseKT	0.7670	0.7092	0.4396	0.7620	0.7477	0.4201	0.7006	0.6557	0.4727	0.8395	0.7718	0.3959
SimpleKT	0.7709	0.7209	0.4372	0.7583	0.7426	0.4212	0.7048	0.6573	0.4730	0.8408	0.7734	0.3963
DKT	0.7591	0.7166	0.4333	0.7288	0.7292	0.4275	0.6589	0.6289	0.4771	0.8029	0.7433	0.4156
HD-DKT	0.7633	0.7195	0.4320	0.7324	0.7344	0.4245	0.6630	0.6321	0.4747	0.8102	0.7487	0.4121
DKT-GOOD	0.7781	0.7299	0.4254	0.7746	0.7501	0.4152	0.6943	0.6513	0.4719	0.8234	0.7565	0.4057
LPKT	0.7667	0.7188	0.4390	0.7593	0.7427	0.4210	0.6941	0.6549	0.4728	0.8329	0.7679	0.3989
HD-LPKT	0.7706	0.7199	0.4330	0.7622	0.7442	0.4186	0.6953	0.6567	0.4713	0.8337	0.7682	0.3987
LPKT-GOOD	0.7794	0.7241	0.4292	0.7881	0.7632	0.4142	0.7227	0.6733	0.4742	0.8475	0.7746	0.3950
AKT	0.7705	0.7192	0.4349	0.7644	0.7497	0.4199	0.7003	0.6592	0.4746	0.8391	0.7745	0.3968
HD-AKT	0.7727	0.7214	0.4310	0.7710	0.7542	0.4161	0.7054	0.6628	0.4684	0.8429	0.7798	0.3941
AKT-GOOD	0.7874	**0.7381**	**0.4256**	0.7844	0.7571	0.4121	0.7308	0.6723	0.4667	0.8482	0.7783	0.3923
DisKT	0.7732	0.7188	0.4390	0.7598	0.7441	0.4240	0.7011	0.6556	0.4742	0.8405	0.7726	0.3962
HD-DisKT	0.7787	0.7268	0.4314	0.7668	0.7448	0.4196	0.7041	0.6617	0.4764	0.8444	0.7769	0.3946
DisKT-GOOD	0.7851	0.7336	0.4298	**0.7981**	**0.7624**	**0.4041**	0.7301	0.6690	0.4657	0.8505	0.7802	0.3852
FoLiBiKT	0.7710	0.7165	0.4356	0.7640	0.7480	0.4195	0.6995	0.6582	0.4756	0.8399	0.7735	0.3962
HD-FoLiBiKT	0.7778	0.7271	0.4318	0.7698	0.7534	0.4156	0.7048	0.6667	0.4701	0.8448	0.7811	0.3947
FoLiBiKT-GOOD	**0.7888**	0.7352	0.4276	0.7954	0.7623	0.4065	**0.7313**	**0.6794**	**0.4653**	**0.8515**	**0.7826**	**0.3846**

Table 2: Performance comparison of different models on various datasets. The best performances for each metric are denoted in **bold**, and the second is underlined. For AUC and ACC, higher values are better, while for RMSE, lower values are better.

AUC versus HD-AKT's 0.7710 on Assist12, and DisKT-GOOD achieves 0.7851 AUC versus HD-DisKT's 0.7787 on Assist09. Third, GOOD boosts performance regardless of backbone architecture. Whether applied to RNN-based models (DKT, LPKT) or attention-based models (AKT, FoLiBiKT, DisKT), GOOD enables significant improvements across all datasets, highlighting its generalizability. Finally, through the novel *learn-generate-to-predict* paradigm, GOOD enables integrated KT models to achieve new SOTA performance. The gains are more pronounced on larger-scale datasets (Assist12 and Ednet). For instance, FoLiBiKT-GOOD boosts AUC by 3.1% on Assist12, yet gains only 1.2% on Spanish. This can be attributed to the diffusion model's ability to learn more nuanced response representations with sufficient training data (Yang et al. 2023).

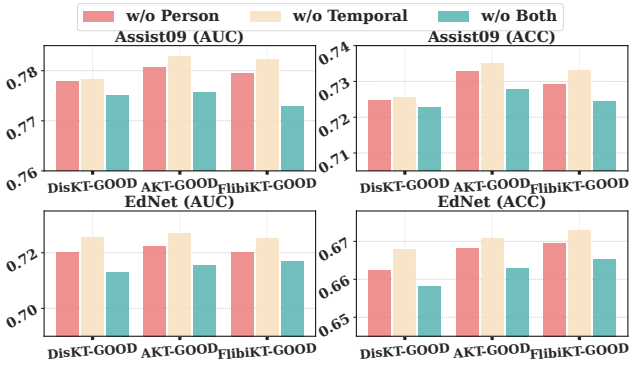

Figure 3: Ablation studies on Assist09 and Ednet datasets to validate the contributions of key components within GOOD.

Ablation studies

Performance under different configurations (RQ2) To validate the effectiveness of our technical innovations, we conduct ablation experiments on the Assist09 and Ednet datasets under three configurations, as shown in Figure 3. First, we evaluate the impact of our person-wise noise scheduling strategy by replacing it with conventional token-wise noise addition (w/o Person). Results show significant performance drops across both datasets, validating that consistent noise levels better preserve the coherent nature of student learning trajectories. Second, we examine the contribution of our knowledge-aware adaptive layer normalization by removing its key operation TemporalConv$(\cdot)$ (w/o Temporal). Performance degrades upon its removal, demonstrating its crucial role in aggregating latent knowledge states locally for generating adaptive parameters. Finally, when both components are removed simultaneously (w/o Both), we observe the most significant performance degradation, confirming that these two innovations work synergistically to facilitate the guided diffusion process.

Impact of Hyperparameter (RQ3) We investigate the impact of hyperparameter λ on model performance across all datasets with AKT-GOOD, DisKT-GOOD, and FoLiBiKT-GOOD. The hyperparameter λ in Eq. 27 controls the relative importance between performance prediction loss and response generation loss. Figure 4 presents the results. Performance typically increases from $\lambda = 0.1$ to an optimal point around $\lambda \in \{0.5, 1.0\}$, then gradually decreases as λ continues to increase. This suggests that while $\mathcal{L}_{\text{GEN}}$ provides valuable guidance, excessive emphasis on reconstruction can interfere with the primary objective $\mathcal{L}_{\text{KT}}$. Notably, even when λ approaches 8, GOOD still boosts predictive performance for AKT, DisKT, and FoLiBiKT. Larger-scale datasets like Ednet and Assist12 show greater tolerance for higher λ values, with peak performance often at $\lambda = 1.0$, while smaller datasets like Spanish demonstrate optimal performance at lower λ values around 0.5. This pattern is likely because larger datasets provide more diverse training exam-

Datasets	Assist09		Ednet	
Metrics	KL↓	L2↓	KL↓	L2↓
AKT	0.5678	0.4359	0.6664	0.4771
AKT-GOOD	0.5347	0.4234	0.6299	0.4597
DisKT	0.5592	0.4332	0.6603	0.4749
DisKT-GOOD	0.5383	0.4212	0.6365	0.4618
FoLiBiKT	0.5695	0.4344	0.6772	0.4782
FoLiBiKT-GOOD	0.5275	0.4183	0.6339	0.4574

Table 3: Quantitative analysis of prediction quality: KL divergence and L2 distance between model predictions and ground truth labels. Lower values indicate better quality.

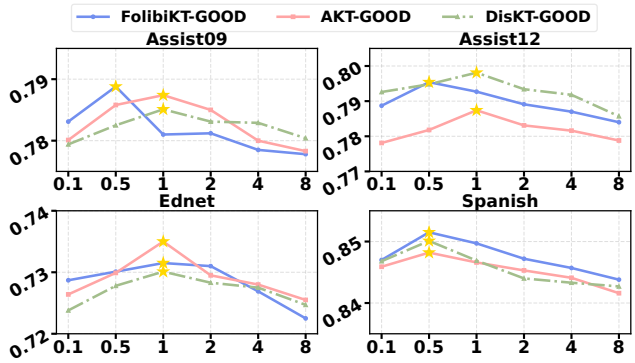

Figure 4: Impact of hyperparameter λ on model performance (AUC) across four datasets.

ples for the generation process. Our analysis reveals that λ demonstrates remarkable stability across most datasets, with GOOD-enhanced models remaining relatively insensitive to variations in the range of 0.1 to 8, thus facilitating practical deployment.

Quantitative Analysis of Prediction(RQ4) To explore how GOOD facilitates more accurate predictions, we conduct a quantitative analysis measuring the distance between model predictions and ground truth labels using two metrics: Kullback-Leibler (KL) divergence (Pérez-Cruz 2008) and L2 distance (Yang and Jin 2006). KL divergence measures the distributional difference between predicted and ground truth distributions, while L2 distance quantifies the Euclidean distance between predictions and ground truth labels. The results in Table 3 demonstrate that GOOD assists AKT, DisKT, and FoLiBiKT in achieving more accurate predictions. Across all backbone models and datasets, the three GOOD-enhanced models consistently reduce both KL divergence and L2 distance. For instance, on Assist09, FoLiBiKT-GOOD achieves a KL divergence of 0.5275 compared to 0.5695 for vanilla FoLiBiKT, and L2 distance decreases from 0.4344 to 0.4183, indicating that generated response representations enable more precise predictions closer to ground truth labels.

Performance-Efficiency Analysis (RQ4) To understand the performance-efficiency trade-offs, we conduct an analysis on the Assist09 dataset, examining relationships between

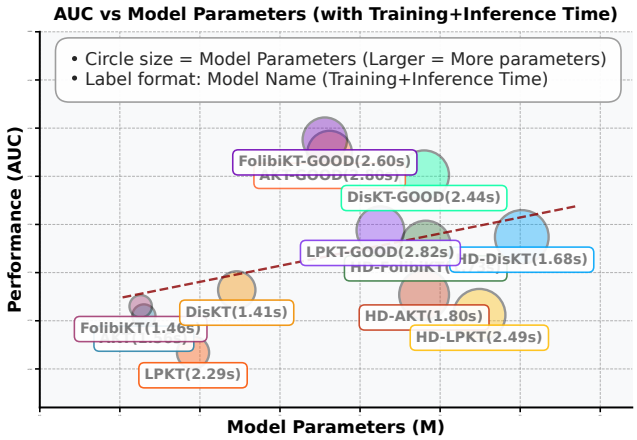

Figure 5: Analysis of model performance trade-offs on Assistments09 dataset. The dashed grey line indicates the general positive trend in the plot.

model performance (AUC), model complexity (parameters), and computational cost (training+inference time). From Figure 5, we have the following findings. First, GOOD exhibits clear advantages over HD in parameter efficiency when applied to LPKT, AKT, DisKT, and FoLiBiKT, achieving higher performance improvements with fewer parameters, making it more practical for resource-constrained environments. Second, the reported time measurements represent combined training and inference time per epoch. GOOD brings remarkable performance improvements while maintaining competitive time complexity compared to HD for all integrated KT models. Finally, this analysis demonstrates that GOOD empowers existing KT models to achieve superior performance while maintaining practical efficiency characteristics, making it suitable for real-world applications.

Conclusion

In this paper, we propose GOOD, a novel plug-in Guided Diffusion Module that introduces a *learn-generate-to-predict* paradigm for KT. Unlike previous approaches that directly predict students' performance from learned latent knowledge states, GOOD assists them further models the problem-solving process of generating target responses, which better aligns with human learning behavior. Through extensive experiments on four benchmark datasets, we demonstrate that GOOD consistently enhances existing KT models and enables them to achieve new SOTA performance across multiple datasets. Furthermore, ablation studies validate the effectiveness of key components within GOOD. The plug-in nature of GOOD enables researchers to easily improve existing KT models within the new *learn-generate-to-predict paradigm*, facilitating broader adoption and comparison studies. In addition, reducing the sampling step while maintaining generation quality is also worthy of further exploration, such as trying consistency models (Song et al. 2023) or flow-based models (Lipman et al. 2022).

Acknowledgments

This work was supported by NSFC (No.62272466, U24A20233), Beijing Municipal Science and Technology Project (No. 2241100004224009), and Big Data and Responsible ArtifcialIntelligence for National Governance, Renmin University of China. We also extend our thanks to Renzhe Xu and Hao Zou for their insightful discussions.

References

Abdelrahman, G.; and Wang, Q. 2019. Knowledge tracing with sequential key-value memory networks. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval*, 175–184.

Abdelrahman, G.; Wang, Q.; and Nunes, B. 2023. Knowledge tracing: A survey. *ACM Computing Surveys*, 55(11): 1–37.

Cao, H.; Tan, C.; Gao, Z.; Xu, Y.; Chen, G.; Heng, P.-A.; and Li, S. Z. 2024. A survey on generative diffusion models. *IEEE transactions on knowledge and data engineering*, 36(7): 2814–2830.

Cheng, K.; Peng, L.; Wang, P.; Ye, J.; Sun, L.; and Du, B. 2024. DyGKT: Dynamic graph learning for knowledge tracing. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 409–420.

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 seventh ACM conference on learning@ scale*, 341–344.

Cui, C.; Yao, Y.; Zhang, C.; Ma, H.; Ma, Y.; Ren, Z.; Zhang, C.; and Ko, J. 2024. DGEKT: A dual graph ensemble learning method for knowledge tracing. *ACM Transactions on Information Systems*, 42(3): 1–24.

Eddy, S. R. 1996. Hidden markov models. *Current opinion in structural biology*, 6(3): 361–365.

Fu, L.; Guan, H.; Du, K.; Lin, J.; Xia, W.; Zhang, W.; Tang, R.; Wang, Y.; and Yu, Y. 2024. Sinkt: A structure-aware inductive knowledge tracing model with large language model. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, 632–642.

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.

Hendrycks, D.; and Gimpel, K. 2016. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33: 6840–6851.

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.

Hwang, J.; and Lee, H. 2025. From knowledge tracing to preference tracing: Capturing dynamic user preferences for personalized recommendation. *Electronic Commerce Research and Applications*, 101527.

Hyvärinen, A.; and Dayan, P. 2005. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6(4).

Kotelnikov, A.; Baranchuk, D.; Rubachev, I.; and Babenko, A. 2023. Tabddpm: Modelling tabular data with diffusion models. In *International conference on machine learning*, 17564–17579. PMLR.

Kuo, M.; Sarker, S.; Qian, L.; Fu, Y.; Li, X.; and Dong, X. 2024. Enhancing deep knowledge tracing via diffusion models for personalized adaptive learning. *arXiv preprint arXiv:2405.05134*.

Lee, W.; Chun, J.; Lee, Y.; Park, K.; and Park, S. 2022. Contrastive learning for knowledge tracing. In *Proceedings of the ACM web conference 2022*, 2330–2338.

Lipman, Y.; Chen, R. T.; Ben-Hamu, H.; Nickel, M.; and Le, M. 2022. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*.

Liu, G.; Zhan, H.; and Kim, J.-j. 2024. Question Difficulty Consistent Knowledge Tracing. In *Proceedings of the ACM on Web Conference 2024*, 4239–4248.

Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Xiong, H.; Su, Y.; and Hu, G. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1): 100–115.

Liu, Z.; Liu, Q.; Chen, J.; Huang, S.; and Luo, W. 2023. simpleKT: a simple but tough-to-beat baseline for knowledge tracing. *arXiv preprint arXiv:2302.06881*.

Loshchilov, I.; Hutter, F.; et al. 2017. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5(5): 5.

Ma, H.; Yang, Y.; Qin, C.; Yu, X.; Yang, S.; Zhang, X.; and Zhu, H. 2024. Hd-kt: Advancing robust knowledge tracing via anomalous learning interaction detection. In *Proceedings of the ACM Web Conference 2024*, 4479–4488.

Maaten, L. v. d.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(Nov): 2579–2605.

Nagatani, K.; Zhang, Q.; Sato, M.; Chen, Y.-Y.; Chen, F.; and Ohkuma, T. 2019. Augmenting knowledge tracing by considering forgetting behavior. In *The world wide web conference*, 3101–3107.

Pandey, S.; and Karypis, G. 2019. A self-attentive model for knowledge tracing. *arXiv preprint arXiv:1907.06837*.

Pérez-Cruz, F. 2008. Kullback-Leibler divergence estimation of continuous distributions. In *2008 IEEE international symposium on information theory*, 1666–1670. IEEE.

Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L. J.; and Sohl-Dickstein, J. 2015. Deep knowledge tracing. *Advances in neural information processing systems*, 28.

Qi, Z.; Bai, L.; Xiong, H.; and Xie, Z. 2024. Not all noises are created equally: Diffusion noise selection and optimization. *arXiv preprint arXiv:2407.14041*.

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.

Shen, S.; Liu, Q.; Huang, Z.; Zheng, Y.; Yin, M.; Wang, M.; and Chen, E. 2024. A survey of knowledge tracing: Models, variants, and applications. *IEEE Transactions on Learning Technologies*.

Song, J.; Meng, C.; and Ermon, S. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Song, X.; Li, J.; Cai, T.; Yang, S.; Yang, T.; and Liu, C. 2022. A survey on deep learning based knowledge tracing. *Knowledge-Based Systems*, 258: 110036.

Song, Y.; Dhariwal, P.; Chen, M.; and Sutskever, I. 2023. Consistency models.

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, J.; Ma, H.; Zhang, M.; Zhang, L.; and Chang, L. 2025. Multi-granularity ensemble interaction graph modeling for knowledge tracing. *Knowledge-Based Systems*, 309: 112834.

Wang, Z.; Feng, J.; Tang, S.; Huang, T.; and Wang, G. 2023. FoLiBiKT: Attention-based knowledge tracing with forgetting curve. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, 4533–4541.

Yang, L.; and Jin, R. 2006. Distance metric learning: A comprehensive survey. *Michigan State Universiy*, 2(2): 4.

Yang, L.; Zhang, Z.; Song, Y.; Hong, S.; Xu, R.; Zhao, Y.; Zhang, W.; Cui, B.; and Yang, M.-H. 2023. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys*, 56(4): 1–39.

Yang, T.; Wu, R.; Ren, P.; Xie, X.; and Zhang, L. 2024. Pixel-aware stable diffusion for realistic image super-resolution and personalized stylization. In *European conference on computer vision*, 74–91. Springer.

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 2023*, 855–864.

Yudelson, M. V.; Koedinger, K. R.; and Gordon, G. J. 2013. Individualized bayesian knowledge tracing models. In *Artificial Intelligence in Education: 16th International Conference, AIED 2013, Memphis, TN, USA, July 9-13, 2013. Proceedings 16*, 171–180. Springer.

Zhang, J.; Shi, X.; King, I.; and Yeung, D.-Y. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, 765–774.

Zhang, K.; Ji, T.; and Zhang, H. 2024. Knowledge tracing via multiple-state diffusion representation. *Expert Systems with Applications*, 255: 124797.

Zhang, Z.; and Sabuncu, M. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. *Advances in neural information processing systems*, 31.

Zhou, Y.; Lv, Z.; Zhang, S.; and Chen, J. 2025. Disentangled Knowledge Tracing for Alleviating Cognitive Bias. *arXiv preprint arXiv:2503.02539*.

Zu, S.; Cai, S.; Tang, W.; Wang, C.; Li, L.; and Shen, J. 2024. GuessKT: Improving Knowledge Tracing via Considering Guess Behaviors. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 12811–12815. IEEE.