Contrastive Cross-Course Knowledge Tracing via Concept Graph Guided Knowledge Transfer

**Wenkang Han, Wang Lin, Liya Hu, Zhenlong Dai, Yiyun Zhou,
Mengze Li, Zemin Liu, Chang Yao, Jingyuan Chen**[*]
Zhejiang University
wenkangh@zju.edu.cn, jingyuanchen@zju.edu.cn

Abstract

Knowledge tracing (KT) aims to predict learners' future performance based on historical learning interactions. However, existing KT models predominantly focus on data from a single course, limiting their ability to capture a comprehensive understanding of learners' knowledge states. In this paper, we propose **TransKT**, a contrastive cross-course knowledge tracing method that leverages concept graph guided knowledge transfer to model the relationships between learning behaviors across different courses, thereby enhancing knowledge state estimation. Specifically, TransKT constructs a cross-course concept graph by leveraging zero-shot Large Language Model (LLM) prompts to establish implicit links between related concepts across different courses. This graph serves as the foundation for knowledge transfer, enabling the model to integrate and enhance the semantic features of learners' interactions across courses. Furthermore, TransKT includes an LLM-to-LM pipeline for incorporating summarized semantic features, which significantly improves the performance of Graph Convolutional Networks (GCNs) used for knowledge transfer. Additionally, TransKT employs a contrastive objective that aligns single-course and cross-course knowledge states, thereby refining the model's ability to provide a more robust and accurate representation of learners' overall knowledge states. Our code and datasets are available at https://github.com/DQYZHWK/TransKT/.

1 Introduction

The popularity of massive open online courses (MOOCs) has expanded access to diverse educational resources, resulting in the accumulation of vast datasets on learner behaviors [Ghosh *et al.*, 2020; Wang *et al.*, 2024; Zhou *et al.*, 2025a]. This has led to the emergence of knowledge tracing (KT) as a critical technique for analyzing learners' behaviors, with the goal of providing personalized learning experiences. KT aims to predict a learner's future performance (*i.e.*, the probability of
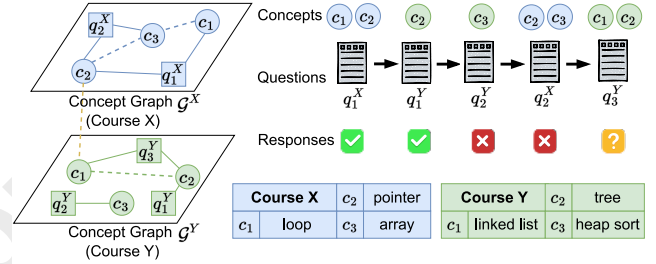
---
[*]Corresponding authors.

Figure 1: An illustrative example of Cross-Course Knowledge Tracing, where each question is associated with one or more concepts. X denotes *C Programming* course, and Y denotes *Data Structure and Algorithm Analysis* course.

correctly answering new questions), by dynamically assessing their knowledge state based on historical interactions.

Most existing KT models [Ghosh *et al.*, 2020; Huang *et al.*, 2023; Liu *et al.*, 2023a; Zhou *et al.*, 2024; Zhou *et al.*, 2025b] are developed based on data from a single course, limiting their ability to capture overall knowledge state across different courses. This paradigm often results in a fragmented understanding, as in practice, learners frequently engage with multiple courses simultaneously [Simamora, 2020; Hu *et al.*, 2023]. The cumulative effect of these varied learning experiences plays a crucial role in shaping a comprehensive knowledge state. In such cases, a single-course focus may lead to unstable and suboptimal predictions, overlooking the interdependencies and transferability of knowledge across different courses. For example, as shown in Figure 1, while Course X and Course Y might not share direct concepts, implicit connections (indicated by the yellow dashed line) between their respective concepts could significantly influence a learner's overall proficiency. This interconnectedness highlights the importance of considering learners' experiences across multiple courses to gain a comprehensive understanding of their knowledge states.

To bridge this gap, we introduce a novel task called cross-course knowledge tracing (CCKT), which aims to enhance knowledge state estimation by leveraging the relationships between learning behaviors across different courses. For example, by considering a learner's historical interactions in both Course X and Course Y, the goal of CCKT is to predict the probability that the learner can correctly answer the

future question, such as q_3^Y.

However, CCKT is not a trivial task and presents significant challenges. Firstly, the connections between concepts are often sparse, especially when comparing concepts from different courses. This scarcity makes it difficult to transfer a learner's proficiency in one concept to potentially related concepts in another course. Secondly, while integrating knowledge states across multiple courses provides a more comprehensive view, it can also introduce noise from unrelated interactions. Therefore, effectively synthesizing learners' knowledge states within and across courses to achieve a robust representation remains another key challenge in CCKT.

To address these challenges, we propose a contrastive cross-course knowledge tracing model via concept graph guided knowledge transfer (denoted as **TransKT**). As shown in Figure 2, TransKT predicts learners' performance on future questions in a course by analyzing their historical learning interactions across multiple courses. Specifically, to address the first challenge, we introduce a cross-course concept graph construction module that establishes implicit links between intra-course and inter-course concepts by an advanced Large Language Model (LLM). Furthermore, based on the constructed cross-course concept graph, we propose a semantic-enhanced knowledge transfer module, which leverages an LLM-to-LM pipeline to summarize and extract rich semantic features, which are then utilized to enhance the performance of Graph Convolutional Networks (GCNs) in cross-course knowledge transfer. Additionally, we derive the learner's knowledge state based on their learning history. Finally, to overcome the second challenge, we propose a cross-course contrastive objective with hybrid hard negative sampling strategy to maximize the mutual information between single-course and cross-course knowledge states. This objective aims to encourage the correlation of learners' knowledge states within and across courses for a more robust knowledge state representation. To demonstrate the validity of our model, we utilize the publicly available PTADisc [Hu *et al.*, 2023] dataset to derive three cross-course KT datasets and use the state-of-the-art KT models for comparison.

The contributions of this paper can be summarized as:

- **Motivation.** This study advances cross-course knowledge tracing, underscoring the critical importance of modeling knowledge acquisition across multiple courses. To the best of our knowledge, the proposed TransKT model represents the first systematic effort to investigate knowledge tracing within a multi-course context, offering novel insights and methodologies for this unexplored yet significant area.

- **Method.** We introduce TransKT, a novel model for cross-course knowledge tracing designed to address the challenges of sparse connections and potential noise in cross-course integration. Firstly, we construct a cross-course concept graph with aligned semantics using an LLM-to-LM pipeline, enabling semantic-based knowledge transfer across courses via graph convolutional networks (GCNs). Secondly, we propose a cross-course contrastive objective that aligns single-course knowledge state representations with their cross-course counterparts, enhancing the robustness and coherence of the synthesized knowledge states. Additionally, as a content-based approach, TransKT allows new questions or concepts to be incorporated without any additional training, offering a significant advantage over traditional ID-based knowledge tracing methods.

- **Experiments.** Extensive experiments on three cross-course knowledge tracing datasets, demonstrating the superiority of TransKT over state-of-the-art baselines.

2 Related Works

2.1 Deep Learning Based Knowledge Tracing

Inspired by deep learning [LeCun *et al.*, 2015], recent knowledge tracing models generally apply deep learning technology. DKT [Piech *et al.*, 2015] stands out as a representative method, employing Long Short-Term Memory to predict the probability of correct responses at each time step. Inspired by memory-augmented neural networks, DKVMN [Zhang *et al.*, 2017] directly predicts learners' knowledge mastery levels based on the values of a dynamic memory matrix, constituting an extension method within the DKT framework. Furthermore, several studies [Ghosh *et al.*, 2020; Chen *et al.*, 2023] have sought to integrate attention mechanisms into KT models following the emergence of the transformer architecture [Vaswani *et al.*, 2017]. The basic idea of these methods is to assign different attention weights to questions in a series of interactions. Additionally, recent studies [Lee *et al.*, 2022; Yin *et al.*, 2023; Zhou *et al.*, 2025b] have considered incorporating contrastive learning into the training of KT models to maintain the stability of knowledge state diagnostics. However, existing KT methods only consider learners' interactions within a single course, overlooking the transfer of knowledge between courses.

2.2 Text-attributed Graph Representation Learning

Text-attributed graph (TAG) representation learning is an emerging field that integrates graph based learning with natural language processing to effectively utilize text attributes in graph-structured data. Recent research has used deep embedding techniques, leveraging pre-trained language models (LMs) like BERT [Devlin *et al.*, 2018] to generate rich node embeddings that encapsulate the semantic depth of text attributes. Approaches such as TextGNN [Zhu *et al.*, 2021] and GIANT [Chien *et al.*, 2021] have demonstrated significant performance improvements by integrating LM-based embeddings with graph neural networks (GNNs). The incorporation of large language models (LLMs) such as ChatGPT [Mann *et al.*, 2020], presents new opportunities for enhancing TAG-related tasks [Lv *et al.*, 2024; Dai *et al.*, 2024; Wu *et al.*, 2024; Guo *et al.*, 2025]. TAPE [He *et al.*, 2023] is a representative method that prompts an LLM for zero-shot classification explanations and then using an LLM-to-LM interpreter to translate these explanations into informative features for enhancing GNN performance. Although these strategies perform well in TAG-related tasks, the application of this idea to cross-course knowledge tracing, where the concept graphs of different courses are independent and the question textual
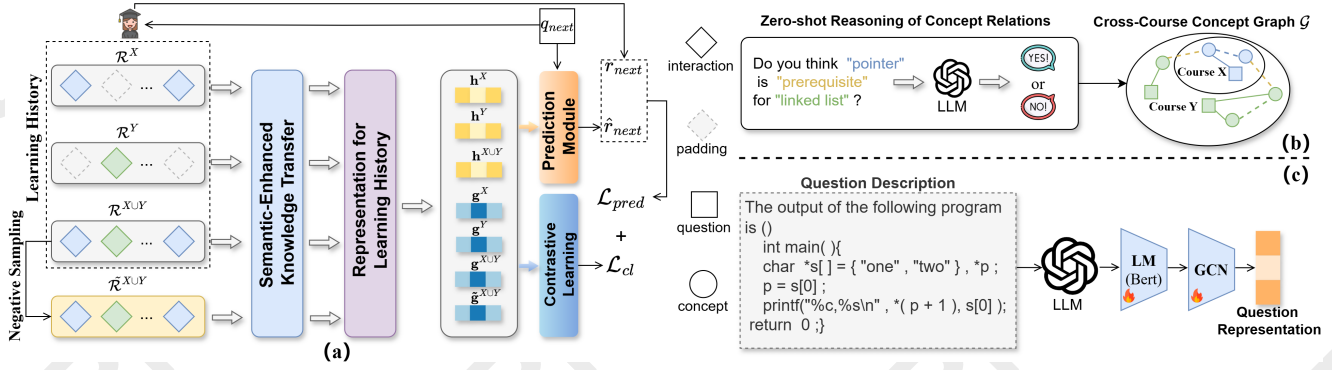
Figure 2: The overview of the proposed TransKT model is shown in (a). The construction of the cross-course concept graph is illustrated in (b). The pipeline of semantic-enhanced knowledge transfer is depicted in (c).

contents of different courses vary greatly in form, has not been explored.

3 Preliminary

Knowledge Tracing (KT) involves tracing learners' knowledge states and predicting their future performance over time. Formally, assuming there is a course with a total of n learners, m questions, and k concepts, which can be denoted as $\mathcal{S} = \{s_1, s_2, \ldots, s_n\}$, $\mathcal{Q} = \{q_1, q_2, \ldots, q_m\}$, and $\mathcal{C} = \{c_1, c_2, \ldots, c_k\}$ respectively. The interaction records of the learner are denoted as $\mathcal{R}$, which is a sequence of tuples (q, r), where $q \in \mathcal{Q}$ and r indicates the binary correctness of the learner's response to the question q. Each question is associated with one or more concepts. To simplify the explanation, we use $\{c\}$ to represent the set of concepts associated with question q. KT aims to predict the probability that a learner will correctly answer the next question $q \in \mathcal{Q}$ by utilizing historical interaction records $\mathcal{R}$.

Cross-Course Knowledge Tracing (CCKT) focuses on predicting a learner's performance on a new question based on their interleaved interaction records from *multiple courses*. In a simplified scenario where learners' interaction records involve two courses, denoted as X and Y, given $\mathcal{R}^X = [(q_1^X, r_1^X), \cdots, (q_{n_X}^X, r_{n_X}^X)]$ and $\mathcal{R}^Y = [(q_1^Y, r_1^Y), \cdots, (q_{n_Y}^Y, r_{n_Y}^Y)]$, representing the interaction records of the learner in courses X and Y, respectively. The interaction records from both courses are merged in chronological order, forming $\mathcal{R}^{X \cup Y} = [(q_1^X, r_1^X), \cdots, (q_1^Y, r_1^Y), \cdots, (q_{n_Y}^Y, r_{n_Y}^Y)]_{n_X + n_Y}$. We use a padding strategy to align the interaction records of $\{\mathcal{R}^X, \mathcal{R}^Y, \mathcal{R}^{X \cup Y}\}$. CCKT aims to predict the probability that the learner will correctly answer the next question $q_{n_X+1}^X$ (or $q_{n_Y+1}^Y$) based on the observed interaction records, which can be formulated as $p(r_{n_X+1}^X=1|\mathcal{R}^X, \mathcal{R}^Y, \mathcal{R}^{X \cup Y}, q_{n_X+1}^X)$ (or $p(r_{n_Y+1}^Y=1|\mathcal{R}^X, \mathcal{R}^Y, \mathcal{R}^{X \cup Y}, q_{n_Y+1}^Y)$).

An example of padding for an interaction sequence of length 5:
$\mathcal{R}^X = [(q_1^X, r_1^X), \text{pad}, \text{pad}, (q_2^X, r_2^X), (q_3^X, r_3^X)]$;
$\mathcal{R}^Y = [\text{pad}, (q_1^Y, r_1^Y), (q_2^Y, r_2^Y), \text{pad}, \text{pad}]$;
$\mathcal{R}^{X \cup Y} = [(q_1^X, r_1^X), (q_1^Y, r_1^Y), (q_2^Y, r_2^Y), (q_2^X, r_2^X), (q_3^X, r_3^X)]$

4 Methodology

In this section, we present the overview of our TransKT model (Shown in Figure 2(a)). Initially, a cross-course concept graph is constructed by predicting concept relations using zero-shot prompting Large Language Models (LLMs) (§4.1). Subsequently, based on the constructed graph, TransKT incorporates a semantic-enhanced knowledge transfer module (§4.2) that utilizes LLMs to extract semantic information as features, which are then utilized to enhance the performance of Graph Convolutional Networks (GCNs) in facilitating cross-course knowledge transfer. Furthermore, TransKT derives the learner's knowledge state based on their learning history(§4.3). Finally, TransKT incorporates a cross-course contrastive objective (§4.4) which maximizes mutual information between single-course and cross-course knowledge states to learn more robust representations for prediction. The prediction and training process of the framework are introduced in §4.5.

4.1 Cross-Course Concept Graph Construction

In modern education, concept graphs (CGs) are widely used as powerful tools for organizing information, offering learners a more intuitive understanding of links between concepts [Ain *et al.*, 2023]. By leveraging these links, knowledge tracing can uncover connections among learning records [Yang *et al.*, 2021; Liu *et al.*, 2020; Yu *et al.*, 2024]. However, in cross-course scenarios, the CGs of different courses are independent of each other. This independence hinders the transfer of knowledge between courses, making it challenging for cross-course knowledge tracing model to identify valuable learning records from other courses. To address this issue, we propose a zero-shot link prediction method to construct a cross-course concept graph $\mathcal{G}$.

As shown in Figure 2(b), the cross-course concept graph, includes two types of nodes: questions and concepts, and two types of links: explicit question-concept links (solid line) and implicit concept-concept links (dotted line). As defined in the Preliminary (§3), the explicit question-concept links are predefined (*i.e.*, the links between q and $\{c\}$). To derive the implicit concept-concept links, we draw inspiration from previous work [Yang *et al.*, 2024] and identify four types of candidate relations (*i.e.*, "Prerequisite_of", "Used_for", "Hy-

ponym_of", "Part_of"). We then design a prompt, pro_r, to leverage the robust zero-shot reasoning capabilities of large language models (LLMs) [Mann *et al.*, 2020] for determining pairwise relations between concepts. The core components of pro_r include the course names, the definition and description of the dependency relation to be predicted, and the query concepts. If a pair of concepts satisfies one of the relation types, an implicit concept-concept link is established in the cross-course concept graph $\mathcal{G}$.

4.2 Semantic-Enhanced Knowledge Transfer

Previous work [Tong *et al.*, 2022] endeavors to integrate semantic information into knowledge tracing. However, in real-world scenarios, the textual content of questions and concepts can often be overly specific or overly abstract, making it challenging to extract meaningful semantic information. Therefore, we propose a semantic-enhanced approach to knowledge transfer in this section. The approach includes: 1) explanation generation with LLMs, 2) semantic feature encoding, and 3) semantic knowledge propagation.

Explanation Generation with LLMs

In educational settings, each question and concept is accompanied by specific textual content, including question descriptions and concept names. The diverse forms of this textual content make it challenging to directly extract effective semantic features with language models. For instance, in computer science course, question descriptions often contain intricate code, while concept names are typically concise and abstract. To align these different types of textual content, as shown in Figure 2(c), TransKT employs an "open-ended" method to query large language models (LLMs) [Mann *et al.*, 2020], leveraging their general knowledge and powerful reasoning capabilities to summarize diverse textual content, which can be formulated as:

$$x_{sum} = \begin{cases} \text{LLM}(pro_q : x_{ori}) & \text{if } x \in \mathcal{Q} \\ \text{LLM}(pro_c : x_{ori}) & \text{if } x \in \mathcal{C} \end{cases}, \quad (1)$$

where x_{ori} denotes the original textual content, x_{sum} denotes the summarized textual content, and $\text{LLM}(\cdot)$ denotes the LLM interface. Prompts pro_q and pro_c are tailored respectively for question descriptions and concept names.

Semantic Feature Encoding

After obtaining x_{sum}, the next step is to convert these text-based outputs into fixed-length semantic features. This transformation facilitates the discovery of semantic relationships within educational content, particularly for questions and concepts across different courses. To achieve this, we fine-tune a smaller language model (LM) to serve as an *interpreter* for the outputs of the LLM. This process extracts the most valuable and relevant semantic features from x_{sum} for knowledge tracing, which can be formulated as:

$$\mathbf{x} = \text{LM}_\theta(x_{sum}) \in \mathbb{R}^D, \quad (2)$$

where $\text{LM}(\cdot)$ denotes a language model based on the transformer structure, such as RoBERTa [Liu *et al.*, 2019]. Here, θ denotes the parameters of the LM, and $\mathbf{x}$ denotes the semantic features of a question or concept.

Semantic Knowledge Propagation

Following the construction of the cross-course concept graph, $\mathcal{G}$, and the extraction of semantic features $\mathbf{x}$ for nodes within $\mathcal{G}$, knowledge transfer across courses is achieved through the utilization of graph convolutional networks (GCNs). Specifically, we utilize GraphSAGE [Hamilton *et al.*, 2017] by stacking multiple GCN layers to encode higher-order neighborhood information. At each layer, the representation of each node is updated by considering both its own semantic feature and those of its neighboring nodes. We denote the feature of node i in the graph as $\mathbf{x}_i$, and the set of its neighbor nodes as $\mathcal{N}_i$. The l-th GCN layer can be expressed as:

$$\mathbf{x}_i^l = \text{ReLU}\left(\frac{1}{|\mathcal{N}_i|} \sum_{j \in \mathcal{N}_i \cup \{i\}} w^l \mathbf{x}_j^{l-1} + b^l\right). \quad (3)$$

After knowledge propagation by GCN, we get the enhanced feature of questions and concepts, denoted as $\mathbf{q}$ and $\mathbf{c}$ respectively. We then combine $\mathbf{q}$ with the representation of the response to indicate interaction as:

$$\mathbf{q}^r = \mathbf{q} + \mathbf{r}, \quad (4)$$

where $\mathbf{r} = x_r \cdot \mathbf{W}_r$, with x_r being a 2-dimensional one-hot vector indicating the correctness of the answer, and $\mathbf{W}_r \in \mathbb{R}^{2 \times D}$ representing a learnable weight parameter.

4.3 Representation for Learning History

The process of knowledge acquisition is inherently heterogeneous due to the diverse characteristics of learners. In the context of knowledge tracing, a learner's historical learning data can serve as a reflection of these distinct traits in the acquisition of knowledge. So, we adopt the attention function to obtain context-aware interaction representations from the learner's historical learning records. The interaction level knowledge state $\mathbf{h}_{t+1}$, achieved after the learner completes the t-th learning interaction, is denoted as:

$$\mathbf{h}_{t+1} = \text{SelfAttention}(Q, K, V),$$
$$Q = \mathbf{q}_{t+1}, K = \{\mathbf{q}_1, \dots, \mathbf{q}_t\}, V = \{\mathbf{q}_1^r, \dots, \mathbf{q}_t^r\}.$$

Then, an average pooling layer $\text{pool}(.)$ is used to represent the entire interaction history, which can be formatted as:

$$\mathbf{g} = \text{pool}(\mathbf{h}_{1:T}). \quad (5)$$

Here, T represents the length of the learning history. The final output, $\mathbf{g} \in \mathbb{R}^D$ is utilized in the subsequent contrastive learning process.

4.4 Cross-Course Contrastive Objective

As mentioned earlier, the single-course knowledge state focuses solely on the learning behaviors related to a particular course, potentially resulting in unstable and suboptimal predictions. Additionally, the cross-course knowledge state offers insights into multiple courses, but may also introduce noise from interactions with other courses. Therefore, it is essential to jointly learn both the learner's single-course and cross-course knowledge state in order to make more accurate predictions. Drawing on the concept of mutual information maximization [Becker, 1996; Hjelm *et al.*, 2019], we propose a cross-course contrastive objective to push together the

Dataset	Java&Python	C&DS	CS&MA
#Records	1,129,999	1,800,066	282,135
#Learners	7,770	12,275	2,431
#Questions	5,734/7,562	11,934/7,624	5,870/1,386
#Concepts	360/364	362/323	359/140

Table 1: CCKT dataset statistics.

single-course and cross-course knowledge state of the same learner. Additionally, inspired by contrastive learning based KT methods [Lee *et al.*, 2022; Yin *et al.*, 2023], we propose a hybrid hard negative sampling strategy designed specifically for cross-course scenarios to further enhance the discriminative capability of the contrastive learning process.

Hybrid Hard Negative Sampling
The hybrid hard negative sampling module contains two strategies, namely the *response flip strategy* and the *interaction replace strategy*.

For the response flip strategy, we randomly flip learners' responses, which can be formatted as:

$$\tilde{\mathcal{R}} = \{(q_i, \tilde{r}_i)\}, \tag{6}$$

where i indicates the index of randomly selected interactions.

For the interaction replace strategy, for each correct interaction, we replace the question with an easier one and mark the response as incorrect, and vice versa. The hard sample generated by this strategy can be formatted as:

$$\tilde{\mathcal{R}} = \{(\mathcal{F}(q_i), \tilde{r}_i)\}, \tag{7}$$

where i indicates the index of randomly selected interactions and $\mathcal{F}(\cdot)$ denotes the application of corresponding replacement operations.

Cross-Course Contrastive Learning
Motivated by maximizing mutual information, we propose a cross-course contrastive objective to push KT model to capture the shared and distinct information across courses. This objective aim to maximize the mutual information between the local (single-course) and global (cross-course) features of the same learner. Taking course X as an example, we seek the single-course knowledge state $\mathbf{g}^X$ to be relevant to the cross-course knowledge state $\mathbf{g}^{X \cup Y}$ but irrelevant to the negative representation $\tilde{\mathbf{g}}^{X \cup Y}$. Therefore, the cross-course contrastive objective for course X is formulated as:

$$\mathcal{L}_{cl}^X = -(\log \mathcal{D}^X(\mathbf{g}^X, \mathbf{g}^{X \cup Y}) + \log(1 - \mathcal{D}^X(\mathbf{g}^X, \tilde{\mathbf{g}}^{X \cup Y}))), \tag{8}$$

where $\mathcal{D}^X$ is a binary discriminator that scores local and global representation pairs through a bilinear mapping function:

$$\mathcal{D}^X(\mathbf{g}^X, \mathbf{g}^{X \cup Y}) = \text{Sigmoid}(\mathbf{g}^{\mathbf{X}} \mathbf{W}_{MI}^X (\mathbf{g}^{\mathbf{X} \cup \mathbf{Y}})^T). \tag{9}$$

Here, $\mathbf{W}_{MI}^X \in \mathbf{R}^{D \times D}$ is a learnable weight matrix. As mentioned in previous work [Cao *et al.*, 2022; Li *et al.*, 2023], the binary cross-entropy loss in Equation (8) serves as an effective mutual information estimator.

4.5 Prediction and Training
Cross-Course Prediction Objective
In cross-course scenarios, when predicting a learner's performance on a new question, we simultaneously utilize the knowledge state from both cross-course and single-course perspectives to form the joint knowledge state. Taking course X as an example, the joint interaction level knowledge state can be formatted as:

$$\hat{\mathbf{h}}_{t+1}^X = \eta \cdot \mathbf{h}_{t+1}^{X \cup Y} + (1 - \eta) \cdot \mathbf{h}_{t+1}^X, \tag{10}$$

where $\eta \in [0, 1]$ is a hyperparameter to balance the influence of the cross-course $\mathbf{h}_{t+1}^{X \cup Y}$ and single-course knowledge state $\mathbf{h}_{t+1}^X$ at timestep $t + 1$. This joint knowledge state $\hat{\mathbf{h}}_{t+1}^X$ is then used to predict the learner's performance on a new question in course X. Specifically, we concatenate $\hat{\mathbf{h}}_{t+1}^X$ with the question representation $\mathbf{q}_{t+1}$ and use the binary cross-entropy loss. The prediction output $\hat{r}_{t+1}^X$ and the loss function $\mathcal{L}_{pred}^X$ are calculated as:

$$\hat{r}_{t+1}^X = \text{ReLU}(\mathbf{W}^X \cdot (\hat{\mathbf{h}}_{t+1}^X \oplus \mathbf{q}_{t+1})), \tag{11}$$

$$\mathcal{L}_{pred}^X = -\sum_t (r_t^X \log \hat{r}_t^X + (1 - r_t^X) \log(1 - \hat{r}_t^X)), \tag{12}$$

where $\mathbf{W}^X \in \mathbb{R}^{D \times 2D}$ represents the learnable weight parameter.

Model Training
During training of TransKT, a regularization factor λ is utilized to balance the prediction loss and contrastive learning loss. The final loss function is:

$$\mathcal{L} = \lambda(\mathcal{L}_{pred}^X + \mathcal{L}_{pred}^Y) + (1 - \lambda)(\mathcal{L}_{cl}^X + \mathcal{L}_{cl}^Y). \tag{13}$$

5 Experiments
We present the details of our experiment settings and the corresponding results in this section. We conduct comprehensive analyses and investigations to illustrate the effectiveness of proposed TransKT model.

5.1 Experimental Setup
Datasets. We further process the publicly available PTADisc dataset [Hu *et al.*, 2023] to obtain three sub-datasets specifically tailored to support the analysis of the CCKT task. These are *Java* and *Python* (Java&Python); *C programming* and *Data Structure and Algorithm Analysis* (C&DS); *C programming* and *Discrete Mathematics* (CS&MA). The statistics for the three CCKT datasets are in Table 1.

Baselines. We compare TransKT with the state-of-the-art methods, including 1) *deep sequential methods*: DKT [Piech *et al.*, 2015], DKT+ [Yeung and Yeung, 2018], IEKT [Long *et al.*, 2021]; 2) *deep memory-aware methods*: DKVMN [Zhang *et al.*, 2017] and Deep_IRT [Yeung, 2019]; 3) *graph based method*: GIKT [Yang *et al.*, 2021]; 4) *attention based methods*: AKT [Ghosh *et al.*, 2020], simpleKT [Liu *et al.*, 2023b], sparseKT [Huang *et al.*, 2023], stableKT [Li *et al.*, 2024]; 5) *contrastive learning based methods*: CL4KT [Lee *et al.*, 2022] and DTransformer [Yin *et al.*, 2023]. To ensure a fair comparison, we perform joint training using the merged cross-course learning records $\mathcal{R}^{X \cup Y}$ for these baselines.

Methods	Java&Python				C&DS				CS&MA			
	Course Java		Course Python		Course C		Course DS		Course CS		Course MA	
	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC	AUC
DKT	0.7988	0.7064	0.7917	0.7128	0.7617	0.6662	0.7906	0.6706	0.7571	0.7002	0.8000	0.8527
DKT+	0.8085	0.7070	0.7939	0.7198	0.7595	0.6674	0.7933	0.6774	0.7562	0.6971	0.7916	0.8500
IEKT	0.7983	0.7132	0.7870	0.7250	0.7567	0.6821	0.7924	0.6871	0.7648	0.7013	0.8077	0.8484
Deep_IRT	0.7964	0.6929	0.7833	0.7058	0.7568	0.6626	0.7872	0.6667	0.7440	0.6643	0.8071	0.8457
DKVMN	0.7988	0.6955	0.7834	0.7077	0.7541	0.6644	0.7828	0.6694	0.7433	0.6769	0.8047	0.8438
GIKT	0.8005	0.7088	0.7880	0.7165	0.7535	0.6708	0.7819	0.6800	0.7456	**0.7033**	0.8014	0.8415
AKT	<u>0.8110</u>	0.7258	<u>0.8035</u>	<u>0.7301</u>	0.7612	0.6783	0.7970	0.6853	<u>0.7679</u>	0.6972	0.8086	0.8447
simpleKT	0.7942	0.7250	0.7933	0.7270	0.7514	0.6825	0.7873	0.6903	0.7664	0.6992	0.8061	0.8523
sparseKT	0.8098	0.7222	0.7976	0.7262	<u>0.7656</u>	<u>0.6872</u>	<u>0.7981</u>	0.6924	0.7512	0.6941	<u>0.8103</u>	<u>0.8551</u>
stableKT	0.7912	0.7194	0.7960	0.7292	0.7577	0.6856	0.7951	0.6850	0.7505	0.6899	0.7998	0.8504
CL4KT	0.8068	<u>0.7260</u>	0.7965	0.7286	0.7630	0.6768	0.7836	<u>0.6931</u>	0.7584	0.6957	0.8035	0.8506
DTransformer	0.8040	0.7152	0.7947	0.7216	0.7601	0.6680	0.7745	0.6863	0.7510	0.6851	0.7998	0.8469
TransKT(Ours)	**0.8341***	**0.7703***	**0.8119***	**0.7440***	**0.7793***	**0.7022***	**0.8214***	**0.7532***	**0.7765***	<u>0.7020</u>	**0.8222***	**0.8629***

Table 2: Performance comparison of TransKT and 12 KT models on three datasets. The best results are in bold, and the second-best results are underlined. * indicates statistical significance over the best baseline, measured by t-test with p-value $\leq$ 0.01.

Methods	Java&Python		C&DS		CS&MA	
	Course Java	Course Python	Course C	Course DS	Course CS	Course MA
AKT*	0.7331(+0.73%)	0.7373(+0.72%)	0.6845(+0.62%)	0.6960(+1.07%)	0.7016(+0.44%)	0.8510(+0.63%)
stableKT*	0.7315(+1.21%)	0.7398(+1.06%)	0.6944(+0.88%)	0.6945(+0.95%)	0.7031(+1.32%)	0.8613(+1.09%)
CL4KT*	0.7344(+0.84%)	0.7361(+0.75%)	0.6872(+1.04%)	0.7015(+0.84%)	0.7008(+0.51%)	0.8570(+0.64%)

Table 3: AUC performance of variant versions of the baseline method across three datasets. * denotes the replacement of each model's original question representation with the one extracted by the proposed semantic-enhanced knowledge transfer module. The values in parentheses indicate the improvement over the original version.

Experimental Settings and Metrics. We use the AdamW optimizer to train all models, fixing the embedding size at 256 and the dropout rate at 0.3. The learning rate and L_2 coefficient are chosen from the sets {1e-3, 1e-4, 1e-5} and {1e-4, 5e-5, 1e-5}, respectively. The hyperparameters η and λ are chosen from the range 0.1 to 0.9 with a step size of 0.1. To ensure a fair comparison, method-specific hyperparameters (*e.g.*, Cognition Space Size for IEKT) were set according to the specifications outlined in their respective papers, while we optimized shared hyperparameters, such as the learning rate and L_2 regularization coefficient, for all baseline methods. In line with existing KT studies, our evaluation metric includes both AUC and Accuracy (ACC). We repeat each experiment 5 times and report the averaged metrics. In addition, we set an epoch limit of 200 and employ an early stopping strategy if the AUC shows no improvement for 10 consecutive epochs.

5.2 Results

Overall Performance

Table 2 shows the performance comparison of our model with other KT models on three CCKT datasets: Java&Python, C&DS, CS&MA. The results reveal several key observations: (1) Our TransKT consistently outperforms the baseline models across all datasets, irrespective of the level of course similarity. Whether the datasets exhibit high course similarity (*e.g.*, Java&Python and C&DS) or significant differences in course topics (CS&MA), TransKT demonstrates superior performance. Specifically, it achieves an average increase of 1.51% and 3.17% over the best baseline model in ACC and

AUC, respectively. This highlights the effectiveness and generalizability of the TransKT. (2) The performance gains of TransKT vary from course to course and are influenced by the distribution of interleaved interaction records. In datasets like C&DS, learners tend to engage more in Course C before Course DS, making the knowledge transfer from Course C to Course DS more beneficial. (3) Among all baseline methods, ranked from weakest to strongest, are deep memory-aware methods, deep sequential methods, graph based method, contrastive learning based methods and attention based methods. AKT performs the best, possibly due to its monotonic attention mechanism modeling forgetting behavior in cross-course learning.

The Impact of Semantic-Enhanced Knowledge Transfer

To further validate the significance of incorporating question and concept semantic features in cross-course scenarios, we select three baselines and enhance them by integrating the cross-course concept graph construction (§4.1) and semantic-enhanced knowledge transfer (§4.2). As shown in Table 3, the results indicate that these baselines perform better on the CCKT dataset than their original versions. Specifically, they achieve average improvements of 0.96% in AUC. This indicates that the design in TransKT for facilitating cross-course knowledge transfer can be seamlessly integrated with other methods, resulting in improved performance on CCKT tasks. Figure 3 visualizes the interaction level attention weights of a single learner at time T (*i.e.*, during interaction $y5$) towards interactions from $T - 9$ to $T - 1$. The AKT model predom-

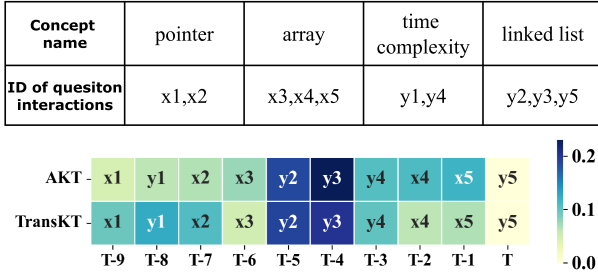| Concept name	pointer	array	time complexity	linked list
ID of quesiton interactions	x1,x2	x3,x4,x5	y1,y4	y2,y3,y5

Figure 3: A representative example illustrating the benefits of semantic-enhanced knowledge transfer.
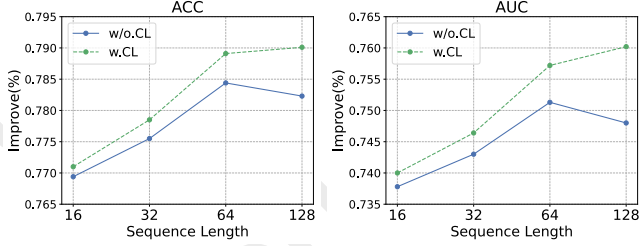

Figure 4: Performance comparison of cross-course contrastive learning objective across different learning sequence lengths.

inantly focuses on interactions $y2$ and $y3$, which share the same concept name ("linked list"), while exhibiting temporal monotonicity in relation to the other interactions. In contrast, TransKT not only emphasizes interactions $y2$ and $y3$ but also extends its focus to interactions $y1$ and $y4$ ("time complexity") within the same course, as well as cross-course interactions $x1$ and $x2$ ("pointer"). This indicates that TransKT effectively links the learner's current interaction with their previous ones based on the semantic similarity of concept names, thereby enhancing knowledge transfer between interactions. These observations underscore the efficacy of the semantic-enhanced knowledge transfer module.

The Impact of Cross-Course Contrastive Learning

The proposed cross-course contrastive learning objective is based on learning history, implying that the performance improvement is correlated with the length of the learning history. To investigate this, we conducted experiments using the Java&Python dataset, further analyzing the scenarios in which this objective applies. As shown in Table 1, the average learning history length in the Java&Python dataset is 145.4. In our experiments, we set four different upper limits for the learning history length: [16, 32, 64, 128] (i.e., truncating learner interaction records within these ranges). As depicted in Figure 4, the benefits of the cross-course contrastive learning objective increased with the upper limit of learning history length, reaching a 1.22% improvement in AUC when the upper limit was set to 128. This highlights the importance of aligning learners' knowledge states in single-course and cross-course in long-term learning scenarios.

Ablation Study

We conduct a comprehensive ablation study on three datasets. We first define the following variations to investigate the im-
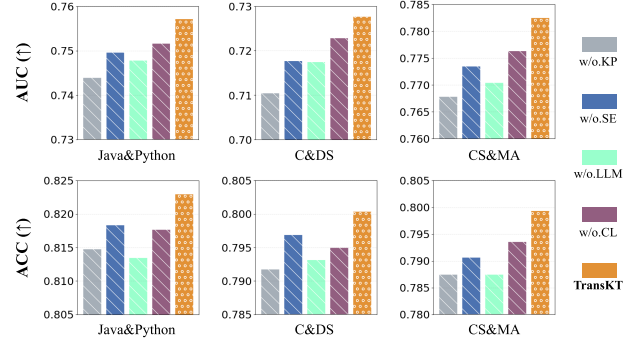

Figure 5: Ablation study.

pact of each component in TransKT: 1) w/o.KP removes the semantic knowledge propagation module. 2) w/o.SE removes the semantic feature encoding from TransKT and use ID-based embeddings for questions and concepts. 3) w/o.LLM directly extracts semantic features from questions and concepts without using LLM for explanation generation. 4) w/o.CL removes the cross-course contrastive objective from TransKT. The results in Figure 5 reveal several observations: (1) All variants suffer relative performance declines across the three datasets on different metrics, demonstrating the contribution of the designed modules in TransKT. Among them, w/o.KP performs the worst, indicating that establishing a cross-course concept graph for knowledge transfer can effectively uncover the correlation between learners' behaviors across different courses. This is fundamental for effective CCKT. (2) The performance of w/o.LLM is even worse than that of w/o.SE, indicating the challenge of directly extracting semantic features without the interpretative capabilities of LLM. In fact, extracting semantic features directly from the original content of questions and concepts performs even worse than using randomly initialized embedding features. (3) By introducing a cross-course contrastive learning objective, TransKT effectively integrates the learning of both single-course and cross-course knowledge states. This approach not only effectively leverages learning information from multiple courses but also helps to mitigate noise from other courses, enhancing the overall performance.

6 Conclusion

In this paper, we introduce TransKT, a novel model for cross-course knowledge tracing (CCKT). TransKT utilizes zero-shot large language model (LLM) queries to construct a comprehensive cross-course concept graph and employs an LLM-to-LM pipeline to enhance semantic features, significantly improving the performance of Graph Convolutional Networks (GCNs) in knowledge transfer. By aligning single-course and cross-course knowledge states through a cross-course contrastive objective, TransKT offers a more robust and comprehensive understanding of learners' knowledge states. Extensive experiments on three real-world datasets demonstrate that TransKT surpasses state-of-the-art KT models in predicting learners' performance across courses.

Acknowledgements

References

[Ain *et al.*, 2023] Qurat Ul Ain, Mohamed Amine Chatti, Komlan Gluck Charles Bakar, Shoeb Joarder, and Rawaa Alatrash. Automatic construction of educational knowledge graphs: a word embedding-based approach. *Information*, 14(10):526, 2023.

[Becker, 1996] Suzanna Becker. Mutual information maximization: models of cortical self-organization. *Network: Computation in neural systems*, 7(1):7, 1996.

[Cao *et al.*, 2022] Jiangxia Cao, Xin Cong, Jiawei Sheng, Tingwen Liu, and Bin Wang. Contrastive cross-domain sequential recommendation. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pages 138–147, 2022.

[Chen *et al.*, 2023] Jiahao Chen, Zitao Liu, Shuyan Huang, Qiongqiong Liu, and Weiqi Luo. Improving interpretability of deep sequential knowledge tracing models with question-centric cognitive representations. *arXiv preprint arXiv:2302.06885*, 2023.

[Chien *et al.*, 2021] Eli Chien, Wei-Cheng Chang, Cho-Jui Hsieh, Hsiang-Fu Yu, Jiong Zhang, Olgica Milenkovic, and Inderjit S Dhillon. Node feature extraction by self-supervised multi-scale neighborhood prediction. *arXiv preprint arXiv:2111.00064*, 2021.

[Dai *et al.*, 2024] Zhenlong Dai, Chang Yao, WenKang Han, Ying Yuan, Zhipeng Gao, and Jingyuan Chen. Mp-coder: Multi-user personalized code generator with explicit and implicit style representation learning. *arXiv preprint arXiv:2406.17255*, 2024.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Ghosh *et al.*, 2020] Aritra Ghosh, Neil Heffernan, and Andrew S Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2330–2339, 2020.

[Guo *et al.*, 2025] Zirun Guo, Shulei Wang, Wang Lin, Weicai Yan, Yangyang Wu, and Tao Jin. Efficient prompting for continual adaptation to missing modalities. *arXiv preprint arXiv:2503.00528*, 2025.

[Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

[He *et al.*, 2023] Xiaoxin He, Xavier Bresson, Thomas Laurent, Adam Perold, Yann LeCun, and Bryan Hooi. Harnessing explanations: Llm-to-lm interpreter for enhanced text-attributed graph representation learning. *arXiv preprint arXiv:2305.19523*, 2023.

[Hjelm *et al.*, 2019] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *International Conference on Learning Representations*, 2019.

[Hu *et al.*, 2023] Liya Hu, Zhiang Dong, Jingyuan Chen, Guifeng Wang, Zhihua Wang, Zhou Zhao, and Fei Wu. Ptadisc: A cross-course dataset supporting personalized learning in cold-start scenarios. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2023.

[Huang *et al.*, 2023] Shuyan Huang, Zitao Liu, Xiangyu Zhao, Weiqi Luo, and Jian Weng. 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*, pages 2441–2445, 2023.

[LeCun *et al.*, 2015] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[Lee *et al.*, 2022] Wonsung Lee, Jaeyoon Chun, Youngmin Lee, Kyoungsoo Park, and Sungrae Park. Contrastive learning for knowledge tracing. In *Proceedings of the ACM Web Conference 2022*, pages 2330–2338, 2022.

[Li *et al.*, 2023] Linjun Li, Tao Jin, Xize Cheng, Ye Wang, Wang Lin, Rongjie Huang, and Zhou Zhao. Contrastive token-wise meta-learning for unseen performer visual temporal-aligned translation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10993–11007, 2023.

[Li *et al.*, 2024] Xueyi Li, Youheng Bai, Teng Guo, Zitao Liu, Yaying Huang, Xiangyu Zhao, Feng Xia, Weiqi Luo, and Jian Weng. Enhancing length generalization for attention based knowledge tracing models with linear biases. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI-24)*, pages 5918–5926, 2024.

[Liu *et al.*, 2019] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[Liu *et al.*, 2020] Yunfei Liu, Yang Yang, Xianyu Chen, Jian Shen, Haifeng Zhang, and Yong Yu. Improving knowledge tracing via pre-training question embeddings. *arXiv preprint arXiv:2012.05031*, 2020.

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

[Liu *et al.*, 2023b] Zitao Liu, Qiongqiong Liu, Jiahao Chen, Shuyan Huang, and Weiqi Luo. simplekt: a simple but tough-to-beat baseline for knowledge tracing. *arXiv preprint arXiv:2302.06881*, 2023.

[Long *et al.*, 2021] Ting Long, Yunfei Liu, Jian Shen, Weinan Zhang, and Yong Yu. Tracing knowledge state with individual cognition and acquisition estimation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 173–182, 2021.

[Lv *et al.*, 2024] Zheqi Lv, Shaoxuan He, Tianyu Zhan, Shengyu Zhang, Wenqiao Zhang, Jingyuan Chen, Zhou Zhao, and Fei Wu. Semantic codebook learning for dynamic recommendation models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 9611–9620, 2024.

[Mann *et al.*, 2020] Ben Mann, N Ryder, M Subbiah, J Kaplan, P Dhariwal, A Neelakantan, P Shyam, G Sastry, A Askell, S Agarwal, et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1, 2020.

[Piech *et al.*, 2015] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J Guibas, and Jascha Sohl-Dickstein. Deep knowledge tracing. *Advances in neural information processing systems*, 28, 2015.

[Simamora, 2020] Roy Martin Simamora. The challenges of online learning during the covid-19 pandemic: An essay analysis of performing arts education students. *Studies in Learning and Teaching*, 1(2):86–103, 2020.

[Tong *et al.*, 2022] Hanshuang Tong, Zhen Wang, Yun Zhou, Shiwei Tong, Wenyuan Han, and Qi Liu. Introducing problem schema with hierarchical exercise graph for knowledge tracing. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 405–415, 2022.

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

[Wang *et al.*, 2024] Ye Wang, Jiahao Xun, Minjie Hong, Jieming Zhu, Tao Jin, Wang Lin, Haoyuan Li, Linjun Li, Yan Xia, Zhou Zhao, et al. Eager: Two-stream generative recommender with behavior-semantic collaboration. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3245–3254, 2024.

[Wu *et al.*, 2024] Tao Wu, Mengze Li, Jingyuan Chen, Wei Ji, Wang Lin, Jinyang Gao, Kun Kuang, Zhou Zhao, and Fei Wu. Semantic alignment for multimodal large language models. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3489–3498, 2024.

[Yang *et al.*, 2021] Yang Yang, Jian Shen, Yanru Qu, Yunfei Liu, Kerong Wang, Yaoming Zhu, Weinan Zhang, and Yong Yu. Gikt: a graph-based interaction model for knowledge tracing. In *Machine learning and knowledge discovery in databases: European conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, proceedings, part I*, pages 299–315. Springer, 2021.

[Yang *et al.*, 2024] Rui Yang, Boming Yang, Sixun Ouyang, Tianwei She, Aosong Feng, Yuang Jiang, Freddy Lecue, Jinghui Lu, and Irene Li. Graphusion: Leveraging large language models for scientific knowledge graph fusion and construction in nlp education. *arXiv preprint arXiv:2407.10794*, 2024.

[Yeung and Yeung, 2018] Chun-Kit Yeung and Dit-Yan Yeung. Addressing two problems in deep knowledge tracing via prediction-consistent regularization. In *Proceedings of the fifth annual ACM conference on learning at scale*, pages 1–10, 2018.

[Yeung, 2019] Chun-Kit Yeung. Deep-irt: Make deep learning based knowledge tracing explainable using item response theory. *arXiv preprint arXiv:1904.11738*, 2019.

[Yin *et al.*, 2023] Yu Yin, Le Dai, Zhenya Huang, Shuanghong Shen, Fei Wang, Qi Liu, Enhong Chen, and Xin Li. Tracing knowledge instead of patterns: Stable knowledge tracing with diagnostic transformer. In *Proceedings of the ACM Web Conference 2023*, WWW '23, page 855–864, New York, NY, USA, 2023. Association for Computing Machinery.

[Yu *et al.*, 2024] Xiaoshan Yu, Chuan Qin, Dazhong Shen, Shangshang Yang, Haiping Ma, Hengshu Zhu, and Xingyi Zhang. Rigl: A unified reciprocal approach for tracing the independent and group learning processes. *arXiv preprint arXiv:2406.12465*, 2024.

[Zhang *et al.*, 2017] Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th international conference on World Wide Web*, pages 765–774, 2017.

[Zhou *et al.*, 2024] Yiyun Zhou, Zheqi Lv, Shengyu Zhang, and Jingyuan Chen. Cuff-KT: Tackling learners' real-time learning pattern adjustment via tuning-free knowledge state-guided model updating, 2024.

[Zhou *et al.*, 2025a] Yiyun Zhou, Wenkang Han, and Jingyuan Chen. Revisiting applicable and comprehensive knowledge tracing in large-scale data. *arXiv preprint arXiv:2501.14256*, 2025.

[Zhou *et al.*, 2025b] Yiyun Zhou, Zheqi Lv, Shengyu Zhang, and Jingyuan Chen. Disentangled knowledge tracing for alleviating cognitive bias. In *Proceedings of the ACM on Web Conference 2025*, pages 2633–2645, 2025.

[Zhu *et al.*, 2021] Jason Zhu, Yanling Cui, Yuming Liu, Hao Sun, Xue Li, Markus Pelger, Tianqi Yang, Liangjie Zhang, Ruofei Zhang, and Huasha Zhao. Textgnn: Improving text encoder via graph neural network in sponsored search. In *Proceedings of the Web Conference 2021*, pages 2848–2857, 2021.