



# Fine-Grained Interaction Modeling with Multi-Relational Transformer for Knowledge Tracing

JIAJUN CUI and ZEYUAN CHEN, School of Computer Science and Technology, East China Normal University

AIMIN ZHOU, School of Computer Science and Technology, Shanghai Institute for AI Education, East China Normal University

JIANYONG WANG, Department of Computer Science and Technology, Tsinghua University

WEI ZHANG, School of Computer Science and Technology, Shanghai Institute for AI Education, East China Normal University

Knowledge tracing, the goal of which is predicting students' future performance given their past question response sequences to trace their knowledge states, is pivotal for computer-aided education and intelligent tutoring systems. Although many technical efforts have been devoted to modeling students based on their question-response sequences, fine-grained interaction modeling between question-response pairs within each sequence is underexplored. This causes question-response representations less contextualized and further limits student modeling. To address this issue, we first conduct a data analysis and reveal the existence of complex cross effects between different question-response pairs within a sequence. Consequently, we propose MRT-KT, a multi-relational transformer for knowledge tracing, to enable fine-grained interaction modeling between question-response pairs. It introduces a novel relation encoding scheme based on knowledge concepts and student performance. Comprehensive experimental results show that MRT-KT outperforms state-of-the-art knowledge tracing methods on four widely-used datasets, validating the effectiveness of considering fine-grained interaction for knowledge tracing.

CCS Concepts: • **Computing methodologies** → **Neural networks**; • **Applied computing** → **Education**; • **Information systems** → **Data mining**;

Additional Key Words and Phrases: Knowledge tracing, multi-relational transformer, user behavior modeling

## ACM Reference format:

Jiajun Cui, Zeyuan Chen, Aimin Zhou, Jianyong Wang, and Wei Zhang. 2023. Fine-Grained Interaction Modeling with Multi-Relational Transformer for Knowledge Tracing. *ACM Trans. Inf. Syst.* 41, 4, Article 104 (March 2023), 26 pages.

<https://doi.org/10.1145/3580595>

This paper was partially supported by the National Natural Science Foundation of China (No. 62072182 and 92270119), the Science and Technology Commission of Shanghai Municipality Grant (No. 21511100101), and the Fundamental Research Funds for the Central Universities.

Authors' addresses: J. Cui and Z. Chen, School of Computer Science and Technology, East China Normal University, No. 3663, North Zhongshan Road, Shanghai 200062, China; emails: cuijj96@gmail.com, chenzyfm@outlook.com; A. Zhou and W. Zhang (corresponding author), School of Computer Science and Technology, Shanghai Institute for AI Education, East China Normal University, No. 3663, North Zhongshan Road, Shanghai 200062, China; emails: amzhou@cs.ecnu.edu.cn, zhangwei.thu2011@gmail.com; J. Wang, Department of Computer Science and Technology, Tsinghua University, Haidian District, Beijing 100084, China; email: jianyong@tsinghua.edu.cn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

1046-8188/2023/03-ART104 \$15.00

<https://doi.org/10.1145/3580595>

## 1 INTRODUCTION

Over the last decade, online education services, such as **massive open online courses (MOOCs)** [17], have flourished due to the rapid development of information technology. These online tutoring systems provide students with various learning materials, including teaching videos, exercises (questions), and so on, and gather amounts of student learning information. These tutoring systems are very important to relieve the imbalance issue of high-quality education resources. However, to realize intelligent online education, immediate and accurate feedbacks to students are crucial as well as the teaching resources. Besides, educators pursue monitoring students' changing knowledge states but lack of recording and quantification. Towards these goals, **knowledge tracing (KT)** [10], which refers to predicting future performance given students' historical question responses, is developed and becomes an indispensable step. The online tutoring systems collect and utilize students' learning information to better serve them by retrieving their hidden knowledge proficiency via the KT tasks. Therefore, students' knowledge states can be traced by educators so that the service of these educational platforms to students and teachers becomes well-directed. Actually, students can recognize their weakness during the learning process in time and have suitable question recommendation according to their knowledge mastery levels. Teachers can also assign personalized learning materials matching the different abilities of students, thus enhancing teaching efficiency. Moreover, some other intelligent educational tasks, such as knowledge concept and learning path recommendation, regard KT as fundamental or auxiliary tasks [1, 19, 48]. As such, computer-aided education and intelligent tutoring systems have taken KT as a key task to be solved [44].

To address the KT problem, many research efforts have been devoted. The earlier studies in this regard are **Bayesian knowledge tracing (BKT)** [33], a probabilistic framework for modeling the generation of student response on questions, and **item response theory (IRT)** based methods [3, 27], combining factors (e.g., question difficulty and student ability) in a logistic function for performance prediction. With the proliferation of neural networks, some recent works pursue more complicated and high-capacity models based on deep learning [24] methods like **deep knowledge tracing (DKT)** [34]. DKT emerges as a paradigm due to its impressive ability of learning dynamic student representations, which conform to the sequentiality of KT tasks. As such, subsequent DKT-derived methods [31, 32, 35–37, 46] mainly focus on **recurrent neural networks (RNN)** and attention mechanisms to model sequences. Benefiting from these advanced techniques, two types of interaction forms, (i) adjacent interaction within students' past question-response sequences and (ii) target-dependent interaction between a target question (to be predicted) and any past question-response pair, are effectively captured to promote the KT performance.

However, most of the existing methods overlook one informative interaction form regarding to different question-response pairs that have a distance larger than one, termed as (iii) non-adjacent interaction form. As shown in Figure 1, the question  $e_4$  is not answered correctly by all the three students. However, the representations of the question-response pair for the three students should be different if we consider student performance on questions  $e_1$  and  $e_2$  that are conceptually relevant to  $e_4$ . Concretely, for the **first student (Stu1)**, we might conjecture that he/she answered  $e_4$  wrongly because he/she does not master the concept "Square Roots" well, according to the right response to  $e_1$  and the wrong response to  $e_2$ . Similarly, we can speculate that the **third student (Stu3)** is not good at the concept "Estimation" and the **second student (Stu2)** does well in both but has an incorrect response when solving  $e_4$ , which might be caused by the knowledge concept combination or the question difficulty. As such, non-adjacent interaction should be considered to achieve better contextual representations for characterizing different students' question-response pairs. This will further facilitate modeling the student knowledge state and personal characteristics for the KT methods.

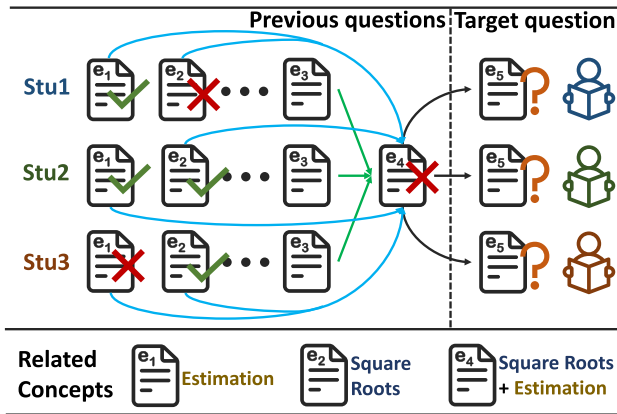


Fig. 1. Illustration of different interaction forms between question-response pairs: green lines for (i) adjacent interaction, black lines for (ii) target-dependent interaction, and blue lines for (iii) non-adjacent interaction.

In the literature, existing methods adequately exploit the first two interaction forms in KT. For RNN-based KT methods, adjacent interaction is modeled implicitly through hidden state transitions. And for attention-based methods, target-dependent interaction is explicitly considered to adaptively aggregate historical response representations. To our best knowledge, there are only a very few existing approaches, i.e., AKT [16] and SAINT+ [36], that learn both adjacent and non-adjacent interactions within student question-response sequences based on self-attention mechanisms [39]. Nevertheless, the realization of interaction modeling is simply based on the inner product of any two question-response pair representations, without differentiating the specific relations between different pairs. In this article, we concentrate on investigating interaction modeling of question-response pairs for boosting the KT performance by considering all three forms of interactions with relation-specific fine-grained modeling.

As revealed in Data Analysis Section 3.2, there are some key observations about cross effects for two question-response pairs from a student: (1) Whether they share some knowledge concepts (knowledge concept sharing) and whether both the two questions are answered correctly or wrongly (student performance consistency) have a positive correlation. (2) There also exists a positive correlation but with complicated patterns between student performance on the first question-response pair and performance on the second question-response pair. In light of this, a key research question raises: how to encode the cross effects for fine-grained interaction modeling within each student's question-response sequence.

To address this, we propose MRT-KT, standing for Multi-Relational Transformer for KT. It can perform fine-grained interaction modeling in transformer networks based on knowledge concepts and student performance. Specifically, the approach first represents questions based on their IDs and associated knowledge concepts. A novel relation encoding scheme is then devised to convert different combinations of student performance and knowledge concept sharing into one type of relation. Thanks to this scheme, any two question-response pairs are assigned by a unique relation. Considering modeling different cross-pair patterns in sequences (i.e., non-adjacent interaction form and target-dependent interaction form), transformer blocks based on multi-relational self-attention and multi-relational target-dependent attention are proposed and stacked to obtain contextualized question-response representations, and the dynamic student representations, respectively. In addition, we introduce relation-specific temporal kernels in these two encoding processes to model different forgetting phenomena for better leveraging multiple relations and

enhancing fine-grained interaction computation. Through such manners, the derived dynamic student representations are utilized with target questions to generate the final performance prediction. Compared with the traditional attention mechanism, which leverages input question, concept, and positional embeddings to calculate the similarity between responses, MRT-KT introduces extra weight matrices and temporal kernels to non-linearly differentiate the underlying relations within students' response sequence, thus explicitly modeling the multiple correlations.

The main contributions of this article are as follows:

**Discovery.** We perform real data analysis and discover that: (1) Sharing knowledge concepts contribute a positive effect to the performance consistency between any two question-response pairs. (2) Student performance on a previous questions-response pair exhibits a positive but complex effect with the performance on a latter questions-response pair, even if they do not share concepts. (3) Effects of both show some complicated patterns when considering different time distances, different datasets, and the composition of concepts and performance.

**Method.** Inspired by data analysis, we propose MRT-KT, a multi-relational transformer network, consisting of a relation encoding scheme to specify unique relation type for any two questions-response pairs, a multi-relational self-attention mechanism to achieve fine-grained interaction modeling, and relation-specific temporal kernels to measure forgetting behaviors of multiple relations.

**Experiment.** Experimental results on four widely-used KT datasets show the effectiveness of MRT-KT over state-of-the-art KT methods. Moreover, a variety of different studies demonstrate that each component of MRT-KT makes significant contributions to the final performance and the interpretability of the model is illustrated.

In what follows, Section 2 briefly introduces the relevant studies on KT and transformer. Then, Section 3 shows the problem formulation and data analysis, supporting the design of our model. In Section 4, the proposed MRT-KT model is introduced in detail. Section 5 analyzes the effectiveness of MRT-KT through comprehensive experiments. Finally, Section 7 concludes this article.

## 2 RELATED WORK

In this section, we review the literature from the following two aspects that are directly relevant to this study, i.e., KT and transformers.

### 2.1 Knowledge Tracing

KT dates back to [10] and has been studied for decades. Earlier research efforts are attributed to the BKT-based methods [21, 33] and the IRT-based approaches [3, 27]. For the BKT-based methods, transition probabilities and emission probabilities are usually adopted to generate students' observed learning interactions with questions. Since posterior probabilities w.r.t. latent binary variables of knowledge concepts could be derived, student knowledge states are understandable. In the IRT-based methods, logistic models are usually leveraged to combine different factors related to student learning interactions, such as student ability, question difficulty, and so on. Due to the linearity of logistic models, the factor contributions could be easily revealed.

However, only in recent years has KT become a focus of research [26], thanks to the introduction of powerful deep learning technologies [24]. In particular, the seminal work DKT [34] is the first deep learning model that uses RNN to recurrently learn student question-response sequences. Compared with standard sequence modeling, student response performance (i.e., answer correctly or wrongly) associated with each question is taken as input as well. Inspired by DKT, some straightforward variants are developed. For example, student clustering information is exploited to enhance input representations [29] and multiple types of information related to

forgetting are used to update hidden states in DKT. The above models exhibit better performance than conventional methods but sacrifice interpretability.

To empower the deep learning based KT methods with interpretability, attention mechanisms are incorporated into interaction modeling between a target question and each of past question-response pairs for KT [16, 31, 32, 37]. The calculated attention weights denote the relative effect of past question-response pairs on the target question. Besides, RNN-based methods also show strong competitiveness. For example, previous work [30] utilizes three different types of temporal information to model forgetting behaviors. LPKT [35] considers both learning and forgetting processes in the hidden state transition. However, as discussed previously, most of these studies neglect the interaction modeling within students' historical sequences, let alone the fine-grained relations addressed in this article.

In addition, some approaches employ other information (e.g., side information) to benefit KT. For example, Chen et al. [4] enhanced the KT loss by proposing a partial-order loss that involves prerequisite knowledge concept relations. The studies [32, 37] utilize textual information of questions to strengthen their embeddings. Besides, Zhou et al. [50] introduced educational context features from three aspects, i.e., home, school, and person. Wang et al. [40] utilized cross-effect temporal information between different concepts. The study [18] effectively leverages the learning and forgetting curves to model student learning behaviors. Recently, AdaptKT [7] is proposed to use student information from other domains to assist in training KT models in the target domain. This is empirically proved to be effective to achieve domain adaption for KT. In contrast to the above studies, this article focuses on the basic data setup of KT and is in parallel to the above approaches.

## 2.2 Transformers

Early studies in transformers are for machine translation [39], which applies attention mechanisms to aggregating word-level representations to form sentence representations. Due to its powerful ability, a large wave of transformer has been continuing in the **natural language processing (NLP)** field. The work [12] improves transformers by sharing transition functions among each layer. Dai et al. [11] considered preserving semantic information from the last segment and proposed a segment-level recurrence technique. The study [23] enhances the efficiency of transformers from both temporal and spatial aspects by adopting hashing mechanisms and reversible residual layers. Kenton et al. [13] presented the far-reaching **Bidirectional Encoder Representations from Transformers (BERT)**, which introduces two unsupervised pre-training language tasks to empower contextualized word representations. A variety of downstream language models finetuned with BERT reach relatively great performance [41, 43].

Studies in the field of recommender systems also utilize transformers in a wide range, especially in sequential recommendation due to the similar sequential structure with language sentences. The pioneering study [20] directly transfers self-attention and point-wise feed-forward layers within historical items. Following the framework of BERT, Sun et al. [38] proposed BERT4REC that predicts masked items by context to strengthen item representations. Other studies like [5, 47] use transformer-based models to capture the sequential signals underlying users' behavior sequences for recommendation.

Recently, transformers have raised extensive attention in the **computer vision (CV)** domain. The work [14] presented **Vision Transformer (ViT)**, splitting images into chunks to form a sequential scheme adapted to the transformer structure. Carion et al. [2] viewed object detection as a direct set prediction problem, adopting a transformer encoder-decoder architecture. Analogously, the recent study [49] treats semantic segmentation as a sequence-to-sequence prediction task. Overall, because of the comprehensive representative capacity, transformers have gained achievements in multiple fields.

The most relevant studies to our work are the KT methods that take student historical responses as prior information and use transformers for sequential modeling. Different from the self-attention mechanism for computing interactions between any two question-response pairs in a sequence, several KT methods [31, 32, 37] only consider target-dependent attention mechanisms to predict the correctness. The exceptions are AKT [16] which employs transformers to encode a historical response sequence with a monotonic attention technique, and SAINT+ [36] which implements an encoder-decoder structure to model non-adjacent interactions. This article goes deep into interaction modeling (with transformers as backbones) by considering fine-grained multi-relations, motivated by data analysis on real KT datasets.

### 3 PRELIMINARY

This section first formulates the studied KT problem. Afterward, data analysis is conducted to show correlations between different question-response pairs w.r.t. knowledge concepts and student performance.

#### 3.1 Problem Formulation

Suppose, we have a student set  $\mathcal{U}$ , a question set  $\mathcal{E}$ , and a knowledge concept set  $\mathcal{C}$ . For student  $u$  ( $u \in \mathcal{U}$ ), we denote the corresponding question-response history up to time step  $t-1$  as  $\mathcal{X}_t^u = \{x_1=(e_1, C_1, r_1, T_1), \dots, x_{t-1} = (e_{t-1}, C_{t-1}, r_{t-1}, T_{t-1})\}$ , indicating each question-response pair is a quadruple with the user index omitted. Take  $x_1$  as an example,  $e_1$  ( $e_1 \in \mathcal{E}$ ) denotes a question and  $C_1$  ( $C_1 = \{c_1, \dots, c_{|C_1|}\}$ ) is its knowledge concept set satisfying  $C_1 \subset \mathcal{C}$ .  $r_1$  equals 1 if the student answers the question correctly, and otherwise 0.  $T_1$  means the timestamp when the response is generated. Given this, the aim of the KT task is to predict the response  $r_t$  to the target question  $e_t$  at time step  $t$  and trace the knowledge states. Under this circumstance, MRT-KT is intended to learn a prediction function as follows:

$$\hat{r}_t = f(\mathcal{X}_t, e_t, C_t, T_t | \Theta),$$

where  $\Theta$  is the trainable parameters of the model. It is worth noting that we assume the question-concept mapping is available in the KT task. This knowledge concept information benefits KT and has been leveraged by most of the current KT methods [16, 25, 35, 40]. Current mainstream online learning platforms/datasets also provide well-annotated question-concept mapping, making this information easy to fetch [8, 15, 42]. To illustrate the model structure and inference procedure clearly, we summarize the key mathematics notations in Table 1, which are used throughout this article. Basically, we bold upper case letters to denote matrices and bold lower case letters to denote vectors, respectively.

#### 3.2 Data Analysis

We conduct data analysis on four widely-used datasets, the details of which are summarized in Table 2. We aim at dictating whether cross effects (correlations) exist between different question-response pairs within the same user sequence. We consider two main types of information used in KT for consideration, i.e., knowledge concepts and student performance. The initial intuition is that a student is more likely to answer a question correctly if he/she has already answered some questions correctly with some shared knowledge concepts (i.e., concept sharing), and vice versa.

To measure the cross effects, we use the Phi coefficient for two given binary variables defined as follows:

$$\phi = \frac{n_{11}n_{00} - n_{01}n_{10}}{\sqrt{n_{0\cdot}n_{1\cdot}n_{\cdot 0}n_{\cdot 1}}},$$

where  $n_{11}$  means the count that the two variables both take value 1 while  $n_{00}$  denotes the count that the value 0 is taken by both. And other symbols could be understood analogously. For any two

Table 1. Key Mathematical Notations

Notations	Description
$u$	student
$e$	question
$c$	knowledge concept
$r$	response correctness
$T$	timestamp
$t$	time step
$x$	interaction
$d$	dimension of latent representations
$a_{ij}, b_{ij}$	attention weight and temporal effect bias w.r.t. interaction $x_i$ to $x_j$
$\kappa_{R_{ij}}$	temporal kernel of relation type $R_{ij}$
$\omega_{R_{ij}}, \beta_{R_{ij}}$	parameters in temporal kernel $\kappa_{R_{ij}}^1$
$w_{R_{ij}}$	trainable adapter to linearly combine $a_{ij}$ and $b_{ij}$
$\alpha_{ij}$	final fine-grained attention weight w.r.t. interaction $x_i$ to $x_j$
$R_{ij}$	encoded relation type w.r.t. interaction of $x_i$ to $x_j$
$\mathbf{e}$	question embedding
$\mathbf{c}$	concept embedding
$\mathbf{x}$	question representation (with concept information)
$\mathbf{q}, \mathbf{k}, \mathbf{v}$	query, key and value vectors in transformer
$\bar{\mathbf{x}}_t$	encoded historical sequence representation up to time step $t$
$\mathbf{u}_t$	encoded student representation at time step $t$
$\mathbf{W}$	trainable matrix in attention calculation
$\mathbf{M}$	trainable matrix in attention aggregation

question-response pairs with the same time step interval, we analyze the following three types of correlations to verify the existence of cross effects:

- (1) Knowledge concept sharing versus student performance consistency. Here, we use  $n_{11}$  to denote the count of two pairs that simultaneously satisfy knowledge concept sharing (value 1) and student performance consistency (value 1). Concretely, for two pairs of question-response pairs, knowledge concept sharing means that two questions share at least one common concept and student performance consistency indicates that the two questions are both answered correctly or both not. This value represents a cross effect that how much **knowledge concept mastery** contributes to correctly answering future questions.
- (2) Student performance on the first question versus the performance on the second question, when the two questions do not share any knowledge concept. Here, we use value 1 to denote the user answers the question correctly. As such,  $n_{11}$  indicates the count of two question-response pairs satisfying that their corresponding questions are both answered correctly. This value represents a cross effect that how much a student's **learning state** contributes to correctly answering future questions. Specifically, learning states refer to a student's personal learning property, e.g., their ability to solve questions or serious attitude to do exercises.
- (3) Student performance on the first question versus the performance on the second question, when the two questions share at least one common knowledge concept. Different from the second type, the third type requires concept sharing. This value represents a comprehensive cross effect including both knowledge concept mastery and learning state factors. It reflects

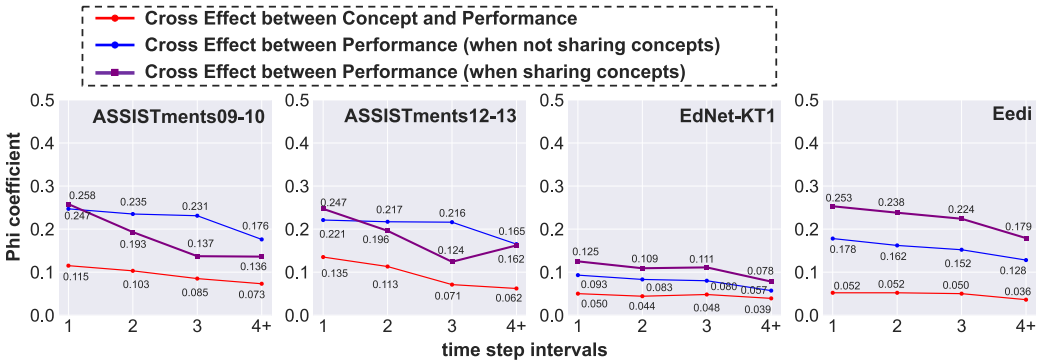


Fig. 2. Data analysis of cross effect for two question-response pairs. The x-axis shows the time step intervals between two pairs.

how much past right responses contribute to correctly answering future questions sharing the same concepts.


It is worth noting that the differentiation of the second and the third types are used to reveal the complicated patterns about cross effects. The overall average results of the coefficients are depicted in Figure 2, where the red lines correspond to the first cross effect, the blue lines for the second, and the purple lines for the third. From this figure, we have the following discoveries:

- For the first cross effect, knowledge concepts and student performance have a positive correlation for different pairs, confirming to the expectation that knowledge mastery affects responses.
- For the second cross effect, performance on the first question and performance on the second question also exhibit a positive correlation, even the two questions do not share at least one concept. This reflects that the learning states of students have continuity to a certain extent.
- The correlation values of the first two cross effects generally become smaller along with the increase of time step intervals between pairs, which adheres to the intuition.
- For the third cross effect that combines both concept mastering and learning states, the correlation values show some complicated patterns when considering different time distances, different datasets, and the composition of concept and performance. It is not only a simple superposition of the first two effects that respectively reflect knowledge mastery and learning state.

We truncate one student’s learning sequence of length 10 and show it in Figure 3. The left red arrow indicates that his/her past most correct responses on the concept “Exponents” help rightly answer a new “Exponents” question, attributed as the knowledge mastery effect. The right blue arrow implies the student retains a good learning state to solve “Divisibility Rules” questions after he/she correctly answers “Exponents” questions, even though these two concepts are almost unrelated. The purple arrow shows the complicated mixed effect that both the knowledge mastery and learning state are unable to help correctly answer new “Divisibility Rules” questions.

The above discoveries suggest there are different types of relations between historical question-response pairs. And different relations have different and complicated impacts on future responses. This motivates us to investigate fine-grained interaction modeling between any two pairs within a student response sequence, which is beneficial to obtain more accurate contextualized representations for each pair. Based on the representations, dynamic user representations could be finally obtained for pursuing better KT performance.





Timeline	Knowledge Concept	Correct
00:00:00	Exponents	✓
00:00:31	Exponents	✗
00:00:57	Exponents	✓
00:01:24	Exponents	✓
00:01:50	Exponents	✓
02:29:30	Divisibility Rules	✓
02:30:07	Divisibility Rules	✓
02:30:27	Divisibility Rules	✓
02:30:54	Divisibility Rules	✗
02:37:27	Divisibility Rules	✗

Annotations:

- cross effect 1 knowledge mastery (red arrow pointing to Exponents rows)
- cross effect 2 learning state (blue arrow pointing to Divisibility Rules rows)
- cross effect 3 complicated mixture (purple arrow pointing to Divisibility Rules rows)

Fig. 3. A learning segment of one student in ASSISTments12–13.

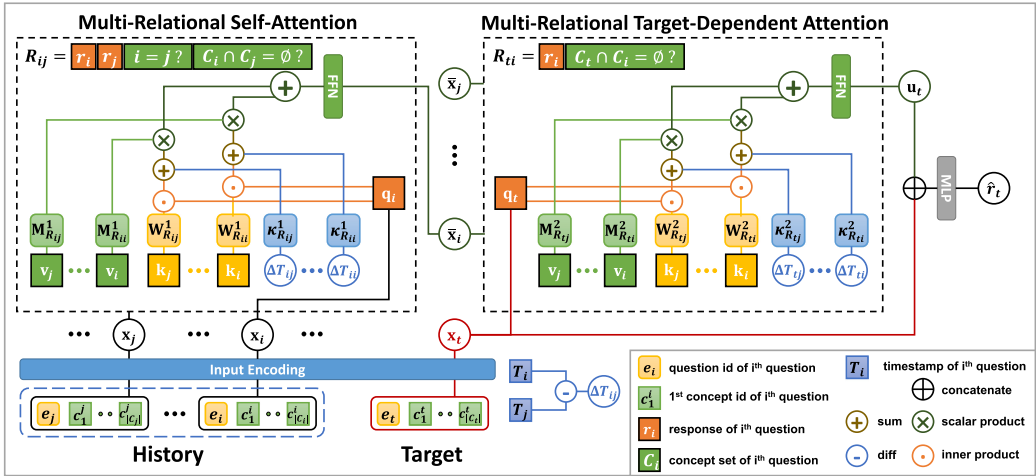


Fig. 4. Overview of the MRT-KT method. The mainly used shapes are explained in the bottom right corner.

## 4 METHODOLOGY

**Overview:** The architecture of the proposed MRT-KT model is depicted in Figure 4. The main aim of MRT-KT is to effectively carry out fine-grained interaction modeling between question-response pairs by differentiating cross effects. To realize this, given input encoding of question-response pairs, MRT-KT proposes a relation encoding scheme to determine a specific relation for any two pairs. The relation is incorporated into multi-relational self-attention for obtaining contextualized question-response representations and multi-relational target-dependent attention for aggregating an overall dynamic student representation. Relation-specific temporal kernels are additionally applied to both of them to model the forgetting behavior. In what follows, we illustrate MRT-KT in detail.

### 4.1 Input Encoding

Input encoding centers on constructing question-response representations, which are taken as input to MRT-KT. Take the  $i$ th question-response pair  $x_i = (e_i, C_i, r_i, T_i)$  in a user question-response

sequence as an example. Assume the embeddings of  $e_i$  and  $c_k$  ( $c_k \in C_i$ ) as  $\mathbf{e}_i \in \mathbb{R}^{\frac{d}{2}}$  and  $\mathbf{c}_k \in \mathbb{R}^{\frac{d}{2}}$ , respectively, where  $d$  is the specified embedding dimension. Then the question representation  $\mathbf{x}_i$  is given by

$$\mathbf{x}_i = \left[ \mathbf{e}_i \oplus \sum_{c_k \in C_i} \mathbf{c}_k \right], \quad (1)$$

where  $\oplus$  denotes concatenation. Different from the current mainstream KT methods considering performance information (i.e., correct or not) in initial embedding layers, we use it to perform fine-grained interaction modeling later. Note that we have also considered adding performance information to input in MRT-KT, but it does not yield improvements.

## 4.2 Relation Encoding Scheme

The objective of the relation encoding scheme is to dictate a specific relation for any two pairs (e.g.,  $x_i$  and  $x_j$ ). This is a crucial step towards fine-grained interaction modeling. Specifically, given knowledge concepts and student performance of the two pairs, a 4-bit binary code and a 2-bit binary code are defined to generate the relation index for multi-relational self-attention and multi-relational target-dependent attention, respectively.

For the 4-bit binary code, as shown in the upper left of Figure 4, the first bit encodes the student response to question  $e_i$ , the second bit encodes the response to  $e_j$  of the same student, the third bit denotes whether the indexes of two pairs are equal, and the fourth bit represents whether the two pairs share at least one common concept. Formally, the relation index is mathematically formulated as follows:

$$R_{ij} = (r_i | r_j | i = j | C_i \cap C_j \neq \emptyset)_2, \quad (2)$$

where  $(\ )_2$  converts a binary string into a decimal value. Although the value ranges from 0 to 15 theoretically, only 10 types of relations are available. This is because when  $i$  equals to  $j$ , the values of  $r_i$  and  $r_j$  are constrained to be the same, and  $C_i \cap C_j \neq \emptyset$  always takes value 1.

Similarly, we can dictate the relation index for the 2-bit binary code. Note that the relations for multi-relational self-attention and multi-relational target-dependent attention are totally different because the response to the target question is unknown, so we have

$$R_{ti} = (r_i | C_t \cap C_i \neq \emptyset)_2, \quad (3)$$

for the target interaction  $x_t$  and historical response  $x_i$ . So far, each pair is assigned a specific relation type for further computation.

## 4.3 Multi-Relational Self-Attention

Transformer networks have exhibited strong sequential modeling ability and become backbones of many well-performed deep learning models, such as BERT [13], graph transformer [45], and DETR [2]. The most essential part in transformer networks is self-attention mechanisms, equipped with a strong interaction modeling ability. Building upon transformer networks, MRT-KT aims to incorporate relations gotten from the relation encoding scheme into self-attention.

Following the procedure of self-attention mechanisms, we assume  $\mathbf{x}_i$  corresponds to key, and  $\mathbf{x}_j$  corresponds to both query and value. By convention, MRT-KT executes transformation operations by  $\mathbf{q}_i = \mathbf{W}_Q^1 \mathbf{x}_i$ ,  $\mathbf{k}_j = \mathbf{W}_K^1 \mathbf{x}_j$ , and  $\mathbf{v}_j = \mathbf{W}_V^1 \mathbf{x}_j$ , where  $\mathbf{W}_Q^1$ ,  $\mathbf{W}_K^1$ , and  $\mathbf{W}_V^1 \in \mathbb{R}^{d \times d}$  are linear projections to query, key, and value. From the relation side, fine-grained representation interaction modeling between  $\mathbf{q}_i$  and  $\mathbf{k}_j$  is implemented by

$$a_{ij}^1 = \frac{\mathbf{q}_i^T \mathbf{W}_{R_{ij}}^1 \mathbf{k}_j}{\sqrt{d}}, \quad (4)$$

where  $\mathbf{W}_{R_{ij}}^1$  is a trainable matrix to control the fine-grained cross effect. The above equation could be easily expanded by a multi-head technique to have more capacity. For simplicity, we omit the details.

Moreover, the temporal distance between two pairs is also important for the KT task based on the forgetting curve theory [6]. To measure the temporal effect on fine-grained interaction modeling, we propose learnable temporal kernels, each of which is tailored for one relation. Formally, the temporal effect is given by

$$b_{ij}^1 = \kappa_{R_{ij}}^1(\Delta T_{ij}), \quad (5)$$

where  $\Delta T_{ij} = T_i - T_j$  and the temporal kernel  $\kappa_{R_{ij}}$  is formulated based on a logarithmic function. Specifically,  $\kappa_{R_{ij}}^1$  is defined as follows:

$$\kappa_{R_{ij}}^1(\Delta T_{ij}) = -\omega_{R_{ij}}^1 \log(\Delta T_{ij} + 1) + \beta_{R_{ij}}^1, \quad (6)$$

where  $\omega_{R_{ij}}^1$  and  $\beta_{R_{ij}}^1$  are relation-specific trainable parameters. We should emphasize that the temporal kernels could be generalized to other ones if necessary.

By combining the representation interaction  $a_{ij}^1$  and temporal effect  $b_{ij}^1$ , we obtain the attention weight  $\alpha_{ij}$  centered on the question-response pair  $x_i$ , given by

$$\alpha_{ij}^1 = \frac{\exp\left(a_{ij}^1 \cdot w_{R_{ij}}^1 + b_{ij}^1\right)}{\sum_{j'} \exp\left(a_{ij'}^1 \cdot w_{R_{ij'}}^1 + b_{ij'}^1\right)}, \quad (7)$$

where the trainable parameter  $w_{R_{ij}}^1$  plays a role of rescaling to make  $a_{ij}^1$  compatible with  $b_{ij}^1$ . Given this, the contextualized question-response representation  $\bar{x}_i$  is obtained by

$$\dot{x}_i = \sum_j \alpha_{ij}^1 \cdot \mathbf{M}_{R_{ij}}^1 \mathbf{v}_j, \quad (8)$$

$$\bar{x}_i = \text{LN}(\text{FFN}(\text{LN}(\dot{x}_i + \mathbf{q}_i)) + \text{LN}(\dot{x}_i + \mathbf{q}_i)), \quad (9)$$

where  $\mathbf{M}_{R_{ij}}^1$  is a relation-specific transformation matrix acting on the value side. FFN and LN, respectively, denote the feed-forward network and layer normalization. Through this manner,  $\bar{x}_i$  encodes multiple relations interacted with other pairs.

Just as general transformer networks, the above computational procedure could be regarded as one transformer layer. To further achieve multi-hop interaction modeling, it could be easily extended to multiple transformer layers. Note that the used superscript 1 corresponds to multi-relational self-attention.

#### 4.4 Multi-Relational Target-Dependent Attention

Based on the contextualized representation sequence  $\{\bar{x}_i\}_{i=1}^{t-1}$ , multi-relational target-dependent attention considers the interaction between the target question and any question-response pair within the sequence. The aim of this component is to derive a student representation  $\mathbf{u}_t$  tailored for the target question. The procedure for getting  $\mathbf{u}_t$  is very similar to that of getting  $\bar{x}_i$  except for the following three aspects.

Firstly, unlike self-attention, the target question representation  $\mathbf{x}_t$  is only taken as query and not used as key or value. This is because the response to the target question is unknown and needs to be predicted to fulfill KT. Due to this structure, the number of layers in multi-relational target-dependent attention is fixed to one.

Secondly, as introduced in the section of Relation Encoding Scheme, the relation space is different because a 2-bit binary code is used. Similarly, only four types of relation-specific temporal kernels are kept.

Thirdly, the parameters used in multi-relational target-dependent attention are not shared with multi-relational self-attention. As illustrated in Figure 4, the mathematical symbols with superscript 2 are used.

#### 4.5 Prediction and Training

**Prediction.** The last component of the MRT-KT method generates the response performance prediction. The input to this component is the concatenation of the student representation  $\mathbf{u}_t$  and the target question representation  $\mathbf{x}_t$ . We adopt **Multi-Layer Perception (MLP)** with **Fully-Connected (FC)** layers and ReLU activation functions to correlate each dimension with nonlinear modeling. Formally, the probability  $\hat{r}_t$  of giving a right response to the question is defined as follows:

$$\hat{r}_t = \sigma(\text{FC}(\cdots \text{ReLU}(\text{FC}([\mathbf{u}_t \oplus \mathbf{x}_t])))\)), \quad (10)$$

where  $\sigma$  is a sigmoid function.

**Training.** To train all the learnable parameters of MRT-KT based on a given training dataset, we generate the response prediction for all the occurred students (i.e., extending  $\hat{r}_t$  to  $\hat{r}_t^u$ , where  $u \in \mathcal{U}$ ) and all the occurred responses of each student (i.e., self-regression training style). Given the ground-truth  $r_t^u$ , the overall binary cross-entropy loss is given by

$$\mathcal{L} = \sum_u \sum_t -\left(r_t^u \log \hat{r}_t^u + (1 - r_t^u) \log(1 - \hat{r}_t^u)\right). \quad (11)$$

The detailed training procedure is summarized in Algorithm 1. We adopt a batch-based gradient method. Given a batch of users with question-response sequences, MRT-KT performs multi-relational self-attention and multi-relational target-dependent attention to generate response prediction. Then, the Adam optimizer is leveraged to update the parameters of MRT-KT by referring to the cross-entropy loss.

#### 4.6 Student Knowledge Tracing

KT tasks predicting students' future responses are also required to trace their learning states, i.e., the mastering level of each concept. Methods like DKT only utilizing knowledge information project hidden representations to  $m$  dimensions where  $m$  is the number of knowledge concepts. Such approaches can easily track the concept mastering scores by directly locating the corresponding dimension, which is not trivial for those methods using question information in the input. Similarly to EKT, we replace the question part with all zero vectors in knowledge state retrieval. Concretely, we use the question-excluded input  $\hat{\mathbf{c}} = [\mathbf{0} \oplus \mathbf{c}]$  where  $\mathbf{c}$  is the embedding of the target concept required to trace. The query into the multi-relational target-dependent block, and the target question representation fed into the MLP predictor are then replaced with  $\hat{\mathbf{c}}$  to get the final concept mastering score. Furthermore, MRT-KT also accepts time input to provide knowledge state retrieval at any time.

#### 4.7 Time Complexity Analysis

MRT-KT enables a fine-grained multi-relational framework that introduces a more complicated attention mechanism than the standard one, in order to deeply model students' knowledge states. To specifically figure out such a tradeoff, we make time complexity analysis within a constant level.

Suppose the sequence length is  $l$ , and the number of hidden dimensions is  $d$ . The standard attention costs  $O(3ld^2)$  for query, key, and value linear projections,  $O(l^2d)$  for calculating attentions, and  $O(l^2d)$  for fusing values. The total time complexity is  $O(3ld^2 + 2l^2d)$ . Operations with a lower complexity order like transformation are omitted. Take the multi-relational self-attention as an example. It assigns each response-question pair with a relation type. Firstly, MRT-KT projects query, key, and value for each response, costing  $O(3ld^2)$  as well. Secondly, the representation interaction

**ALGORITHM 1:** Training Procedure of MRT-KT**Input:**

The training set  $\mathcal{B}$ ; the initialized MRT-KT model  $f(\cdot|\Theta)$ ;

**Output:**

The optimized  $f(\cdot|\Theta)$  after training;

- 1: **for** number of training epochs **do**
- 2:   Sample a mini-batch  $B = \{\mathcal{X}_{t+1}^u\} \in \mathcal{B}$  where each historical sequence  $\mathcal{X}_{t+1}^u$  of student  $u$  contains all of his question-response interactions up to time step  $t$ ;
- 3:   **for**  $\mathcal{X}_{t+1}^u = \mathbf{x}_{1:t}^u$  in the mini-match  $B$  **do**
- 4:     # Omit user index below
- 5:     Obtain input interaction embedding  $\mathbf{x}_{1:t}$  by input encoding (Equation (1));
- 6:     Perform **Multi-Relational Self-Attention** with  $\mathbf{x}_{1:t}$  to attain the question-response representation  $\bar{\mathbf{x}}_{1:t}$ ;
- 7:     Perform **Multi-Relational Target-Dependent Attention** with  $\mathbf{x}_{2:t}$  as query,  $\bar{\mathbf{x}}_{1:t-1}$  as key and value to attain the final student representations  $\mathbf{u}_{2:t}$ ;
- 8:     Predict correct response probability with  $\mathbf{u}_{2:t}$  and  $\mathbf{x}_{2:t}$  by MLP (Equation (10)) to generate the estimated values  $\hat{r}_{2:t}$ ;
- 9:     Calculate binary cross-entropy loss  $\mathcal{L}$  with  $\hat{r}_{2:t}$  and  $r_{2:t}$  by Equation (11) and accumulate the gradients w.r.t. the learnable parameters  $\Theta$ ;
- 10:   **end for**
- 11:   Update parameter weights  $\Theta$  by the Adam optimizer with the accumulated gradients;
- 12: **end for**
- 13: **return** The optimized  $f(\cdot|\Theta)$ .

is calculated (Equation (4)) by the following steps. For each query response, (1) it costs  $O(d^2 + d)$  to calculate the interaction on itself; (2) there are four concept-performance relation types of not-self key responses (e.g., a correct and sharing concept key response) to the query response so that the cost for one type is  $O(d^2 + ld/4)$  in average. Therefore, the total complexity for the query response is  $O(5d^2 + ld + d)$ . Multiplied by the whole length of queries, the total complexity for Equation (4) is  $O(5ld^2 + l^2d + ld)$ . Thirdly, MRT-KT derives attention weights by Equation (5)–(7) using temporal kernels, costing  $O(l^2)$ . Finally, the obtained attention is used for the fusion of values (Equation (8)) with a similar process as the attention calculation, costing  $O(5ld^2 + l^2d + ld)$ . Therefore, omitting operations with lower complexity orders, the total complexity of multi-relational self-attention is  $O(13ld^2 + 2l^2d)$ . Similarly, the target-dependent attention costs  $O(11ld^2 + 2l^2d)$ , which removes the calculation of self-attention.

Due to the real-world setting that  $d$  and  $l$  usually have the same magnitude order (50–200), we assume there is  $n = l = d$ . Then MRT-KT costs  $O(15n^3)$  (or  $O(13n^3)$ ), compared with the standard one's,  $O(5n^3)$ . Therefore, MRT-KT only takes an increase in time complexity within a small constant level (2.6x–3x), which could be easily diluted by other fixed operations in KT models, but brings in a significant improvement over the traditional transformer with the standard attention mechanism. We test the inference speed of MRT-KT on a single i7-8700K CPU. It averagely takes 7–8 ms to trace one student; meanwhile, a two-layers transformer costs 3–4 ms in the same environment.

## 5 EXPERIMENTS

In this section, we conduct a variety of experiments and present detailed results to answer the following essential research questions:

- Q1:** What are the prediction results of MRT-KT compared with other strong and recent KT methods?
- Q2:** How do the main components within MRT-KT significantly contribute to the KT performance?
- Q3:** Is the fine-grained multi-relational interaction modeling able to learn the effects of different relation types and trace student knowledge states effectively?

To answer these questions, we first provide details of the experimental setup, including the used datasets, evaluation methods, compared baselines, and implementation details. Afterward, we present the KT performance to compare MRT-KT with different baselines. An ablation study is then carried out to verify the effectiveness of each component. After that, hyperparameter analysis is performed to show how the model performance is affected by them. Furthermore, we present experiments to demonstrate the model interpretability, including interaction visualization and temporal kernel analysis. A case study is finally adopted to probe the model's capability to selectively focus on historical sequences and trace knowledge states by giving three students' learning examples.

## 5.1 Experimental Setup

**5.1.1 Dataset.** We adopt the following four widely-used KT datasets for performance evaluation, which are publicly available for experiment reproduction.

- **ASSISTments09–10** [15].<sup>1</sup> This dataset is gathered during 2009 to 2010 from the online tutoring system ASSISTments, which teaches and accesses students in mathematics. We choose the version of file *skill\_builder\_data\_corrected\_collapsed.csv* and take the *skill\_id* field as concepts instead of *skill\_name*. This is because *skill\_name* only presents one concept name for each question. Since there are no absolute timestamps indicating when students answer questions, we follow the previous study [40] to determine the timestamp of a response. Specifically, for each question in a question-response sequence, we sum up all its previous response duration time as the timestamp.
- **ASSISTments12–13** [15].<sup>2</sup> It is another dataset from the same platform, ranging from 2012 to 2013. This dataset includes timestamps and has one knowledge concept for one question.
- **EdNet-KT1** [8].<sup>3</sup> EdNet is a large-scale dataset collected by the artificial intelligence tutoring system Santa, consisting of four datasets named KT1, KT2, KT3, and KT4 with different extents. We choose the KT1 dataset.
- **Eedi** [42].<sup>4</sup> This dataset is collected during two school years (2018–2020) of students' answers to mathematics questions from Eedi, a free homework and teaching platform for primary and secondary schools in the UK. We recombine the training and test data used for task 1 and make them suitable for our experimental setup. Moreover, we omit the first two tagged categories of each question (e.g., "Math" because they are coarse-grained) and use the rest as their knowledge concepts. Besides, we directly use response timestamp intervals as the response duration, which is unavailable in this dataset.

For each of the above datasets, we cut every student's response sequence into subsequences with a fixed length of 50. The ones containing less than five responses are removed. Note that

<sup>1</sup><https://sites.google.com/site/ASSISTmentsdata/home/assistance-2009-2010-data>.

<sup>2</sup><https://sites.google.com/site/assistmentsdata/home/2012-13-school-data-with-affect>.

<sup>3</sup><https://github.com/riiid/EdNet>.

<sup>4</sup><https://eedi.com/projects/neurips-education-challenge>.

Table 2. Statistics of the Four Datasets

	ASSISTments09–10	ASSISTments12–13	EdNet-KT1	Eedi
#sequence	8.3 k	67.1 k	2.3 m	447.8 k
#question	5.7 k	53.0 k	12.3 k	27.6 k
#concept	195	265	189	386
#response	113.5 k	2.7 m	95.0 m	19.8 m
#concept per question	1.18	1.00	2.28	2.17

any temporal information, including timestamps and response time is unified in seconds, and any record with a missing field we need is excluded. The statistics of the processed datasets are shown in Table 2.

**5.1.2 Evaluation.** To evaluate model performance, we segment students into training, validation, and test sets for each dataset. The segmentation ratio is set to 8 to 1 to 1. To enhance the statistical significance of the experimental results, we repeat running every method five times with different random seeds and report average results on the test sets. **Area under the curve (AUC)** and **accuracy (ACC)** are used as evaluation metrics, which are commonly used in the KT task.

**5.1.3 Baselines.** We compare MRT-KT with seven typical baseline methods focusing on different aspects in KT.

- **DKT** [34]. A milestone method that applies RNN to KT by capturing hidden knowledge states of each student and shows improvements over BKT. Only concepts are taken as inputs.
- **DKT-Forget** [30]. A direct extension of DKT that incorporates information related to forgetting behavior (e.g., time gap) into representations.
- **SAKT** [31]. An attention-based method for KT. It regards target questions as queries and historical question-response pairs as keys and values.
- **EKT-A** [25].<sup>5</sup> An RNN-based method exploring both students’ response records and the text content of the corresponding questions. Here we use the superior attention version EKT-A.
- **AKT** [16].<sup>6</sup> A transformer-based KT method that considers both interactions within a historical question-response sequence and interactions with a target question.
- **HawkesKT** [40].<sup>7</sup> A Hawkes process based method that characterizes temporal cross effects between any historical question and a target question through a mutual-excited intensity function.
- **LPKT** [35].<sup>8</sup> An RNN-based method considering the consistency of the learning and forgetting process. Response time is taken as input.
- **SAINT+** [36]. An attentive method using an encoder-decoder structure consisting of non-adjacent and target-dependent interactions modeling.

We summarize the properties of all the above models and the proposed model MRT-KT in Table 3. As can be seen, all the baseline models do not consider the multiple relations between the interactions of different question-response pairs.

**5.1.4 Implementation Details.** All the experiments are conducted on a Linux server with GPUs of GeForce GTX 1080Ti and the deep learning framework Pytorch. For open-source methods, we

<sup>5</sup><https://github.com/bigdata-ustc/ekt>.

<sup>6</sup><https://github.com/arghosh/AKT>.

<sup>7</sup><https://github.com/THUwancy/HawkesKT>.

<sup>8</sup><https://github.com/bigdata-ustc/EduKTM>.

Table 3. Properties of Different KT Models

Property	Recurrent	Attentive	Temporal	Adjacent	Non-adjacent	Target-dependent	Multi-relational
DKT	✓			✓			
DKT-Forget	✓		✓	✓			
SAKT		✓				✓	
EKT-A	✓	✓		✓		✓	
AKT		✓	✓	✓	✓	✓	
HawkesKT			✓			✓	
LPKT	✓		✓	✓		✓	
SAINT+		✓	✓	✓	✓	✓	
MRT-KT		✓	✓	✓	✓	✓	✓

The fourth to sixth properties correspond to the interaction forms in Figure 1.

duplicate their codes and make a slight modification to fit our tests. For the methods without public codes (DKT, DKT-Forget, SAKT, and SAINT+), we strictly implement the models according to the details in their original papers.

All the methods including MRT-KT are tuned to reach their respective good performance with the following policy. The batch size is fixed to 128 and the hidden dimension size is selected in {32, 64, 128, 256}. The number of layers in relevant models is tuned from 1 to 4. The methods applying dropout techniques use the ratio chosen from 0 to 0.9 with 0.1 as the interval. Multi-head attention-based approaches select the number of heads in {1, 2, 4, 8}. The introduced time scale for temporal methods is selected in {1, 1e-1, 1e-2, 1e-3}. Moreover, we adopt the Adam optimizer [22] for all the methods with the learning rate in {1e-3, 5e-4, 2e-4, 1e-4, 5e-5, 2e-5, 1e-5}, and L2 normalization values in {1e-3, 5e-4, 1e-4, 5e-5, 1e-5, 0}. We also apply the early stopping strategy to halt the training process when the product of ACC and AUC on the validation sets does not peak in the previous 20 epochs.

For MRT-KT, we adopt a 3-layer MLP for response performance prediction. The embedding dimension  $d$  is set to {128, 128, 256, 128} for the four datasets. The numbers of attention heads and attention layers are respectively set to {4, 4, 4, 4} and {1, 1, 1, 1} for both multi-relational self-attention and multi-relational target-dependent attention. The dropout probabilities in the attention calculation and FFN/MLP are set to 0.2. And, the probabilities in input embedding are set to {0.3, 0.2, 0, 0.1} for the four datasets. Analogously, the learning rates are set to {1e-3, 5e-4, 5e-4, 5e-4} and the L2 normalization values are set to {1e-4, 1e-5, 0, 1e-5}.

Considering the fact that textual information is missing in the datasets, for EKT-A, we remove question description embedding components in the experiments. In addition, we introduce extra temporal hyperparameters for methods utilizing time information, which rescale the temporal values for a fair comparison.

## 5.2 Experimental Results

**5.2.1 Overall Performance.** Table 4 shows the overall performance of the baseline methods and ours in four datasets. From a whole perspective, MRT-KT outperforms all the baseline methods and yields improvements compared with the best baseline. Furthermore, the largest increase is varying from 1.3% to 2.0% on the ASSISTments09–10 and the results on the other three are similar, varying from 0.4% to 1.4%.

Among four RNN-based methods (i.e., DKT, DKT-Forget, EKT-A, LPKT), we find that LPKT is generally better than the others. Besides, because of the fusion of temporal information, DKT-Forget shows a slight enhancement than DKT, which also yields strong results compared with all the other baselines. EKT-A also provides considerable performance compared with other RNN baselines, even though it does not utilize textual information.



Table 4. Comparison of All the Adopted Models

Dataset	ASSISTments09–10		ASSISTments12–13		EdNet-KT1		Eedi	
Metric	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
DKT	0.7850	0.7630	0.7259	0.7329	0.6866	0.6866	0.7696	0.7286
DKT-Forget	0.7830	0.7592	0.7301	0.7358	0.6910	0.6880	0.7712	0.7301
SAKT	0.7894	0.7649	0.7206	0.7306	0.6929	0.6879	0.7643	0.7208
EKT-A	0.7896	0.7693	0.7273	0.7301	0.6873	0.6861	0.7668	0.7281
AKT	<u>0.8060</u>	<u>0.7724</u>	<u>0.7592</u>	<u>0.7451</u>	<u>0.7696</u>	<u>0.7286</u>	<u>0.8194</u>	<u>0.7538</u>
HawkesKT	0.8054	<u>0.7738</u>	0.7463	0.7360	0.7518	0.7203	0.7810	0.7374
LPKT	0.7901	0.7693	0.7488	0.7407	0.7597	0.7243	0.7798	0.7321
SAINT+	0.8013	0.7705	0.7503	0.7425	0.7362	0.7184	0.8012	0.7433
<b>MRT-KT</b>	<b>0.8223</b> *	<b>0.7841</b> *	<b>0.7698</b> *	<b>0.7544</b> *	<b>0.7753</b> *	<b>0.7319</b> *	<b>0.8260</b> *	<b>0.7569</b> *
Improv.	2.0%	1.3%	1.4%	1.2%	0.7%	0.5%	0.8%	0.4%

The best result in each column is in bold and the second one is underlined. \* indicates statistically significance (measured by T-test) with  $p \leq 0.01$  over the best competitor.

For attention-based models, a large performance variance is presented. The most straightforward method SAKT gets the worst performance, showing that only considering target-dependent interaction based on attention mechanisms is insufficient. Meanwhile, AKT almost obtains the secondary results, which are only inferior to the proposed MRT-KT, and SAINT+ achieves good performance on the two ASSISTments datasets. This also verifies the necessity of combining multiple interaction forms.

Even though HawkesKT does not apply typical sequential techniques such as RNN or transformers, the powerful temporal modeling with Hawkes process makes it behave relatively well, especially on the ASSISTments09–10 dataset. Other methods using temporal information (i.e., DKT-Forget, AKT, LPKT, MRT-KT, SAINT+) likewise present better performance, verifying that capturing the forgetting behavior is critical for KT.

**5.2.2 Ablation Study.** We conduct ablation experiments to investigate the effects of multiple relations on interaction modeling. Here, **ConR** is short for concept relation shown in the section of Relation Encoding Scheme, and similarly, **PerR** is short for performance relation. **Con** is short for concept. This ablation is to verify the role of concept information in our MRT-KT. Compared to the “ConR”, it also includes the concept embedding at the input and the prediction stage. We use **PerI** to denote encoding student performance in model input as previous KT studies. Besides, we test the contribution of temporal kernels. To validate the effectiveness of fine-grained multi-relations modeling using the proposed transformers, we also implement a vanilla transformer with question, concept, and positional embedding. Moreover, we design two variants of MRT-KT that directly leverage the discoveries in data analysis, to demonstrate the necessity of such multi-relational modeling. The model “coef. MRT-KT” indicates, we replace the whole multi-relational framework by directly multiplying the phi coefficients as weights in the attention calculation in a normal transformer. To be specific, we shift the phi values from  $[-1, 1]$  to  $[0, 1]$  via a sigmoid function with a temperature hyperparameter to tune. Another model “adapt. MRT-KT” refers to setting learnable adaptive scalar parameters to reweight the standard attention of different types of relations. The results are shown in Table 5, based on which we have the following observations:

- Compared with MRT-KT, “- PerR + PerI”, which denotes removing the performance relation in relation encoding scheme and adding performance representations to model input like most of the KT methods do, degrades the performance significantly. This result shows that encoding student performance information into interaction modeling is a more effective

Table 5. Results of the Ablation Study

Dataset	ASSISTments09–10		ASSISTments12–13		EdNet-KT1		Eedi	
Metric	AUC	ACC	AUC	ACC	AUC	ACC	AUC	ACC
- PerR + PerI	0.8003	0.7678	0.7510	0.7437	0.7392	0.7197	0.8034	0.7460
- ConR	0.8086	0.7725	0.7644	0.7497	0.7580	0.7238	0.8076	0.7491
- PerR	0.7529	0.7498	0.7016	0.7113	0.6743	0.6826	0.7554	0.7205
- (PerR + ConR)	0.7443	0.7478	0.6964	0.7002	0.6690	0.6782	0.7449	0.7164
- Con	0.7998	0.7703	0.7562	0.7437	0.7523	0.7214	0.8018	0.7448
- Time	0.8190	0.7782	0.7640	0.7526	0.7704	0.7289	0.8195	0.7521
Transformer	0.7981	0.7667	0.7452	0.7393	0.7301	0.7159	0.7949	0.7399
ceof. MRT-KT	0.7962	0.7661	0.7439	0.7389	0.7320	0.7164	0.7967	0.7402
adapt. MRT-KT	0.8049	0.7711	0.7526	0.7407	0.7482	0.7203	0.8086	0.7458
<b>MRT-KT</b>	<b>0.8223</b>	<b>0.7841</b>	<b>0.7698</b>	<b>0.7544</b>	<b>0.7753</b>	<b>0.7319</b>	<b>0.8260</b>	<b>0.7569</b>

methodology than simply treating performance as input. One reason for this is that traditional question-response embedding regards the same question with different performance as two independent instances, which introduce twice as many embedding parameters and lose the same question information. This makes the model difficult to converge. On the contrary, MRT-KT directly encodes the relations instead of the question-response pair itself.

- By seeing the results of the middle three variants, we find both concept relations and student performance have positive contributions to KT. What’s more, the two factors could be complementary to each other, since “- (ConR + PerR)” suffers from the largest performance drop. In addition, compared with “- ConR”, “- PerR + PerI” drops a bit more, which demonstrates that student performance contributes more than concept relations in MRT-KT. This enlightens us that in student performance prediction tasks, previous response correctness, i.e., learning state, is a key point to be deeply considered.
- The removal of entire question-concept relations in MRT-KT, denoted as “- Con”, also shows degeneration. This suggests that even though the question-concept mapping introduces noises due to human annotation error, it also benefits KT. Besides, the differences between “- Con” and “- ConR” demonstrate the contribution of the knowledge concept embedding.
- MRT-KT is superior to “- Time” with a marginal discrepancy. This reveals that under our model framework, considering time distance with temporal kernels as attention biases in both two types of attention blocks plays a contribution, which also suggests that students hold different forgetting behaviors for different types of relations.
- Compared with the vanilla transformer, MRT-KT also shows a great enhancement even on the two large-scale datasets. This demonstrates that only taking the calculated response similarity as the attention in the original transformer to exploit multi-relations in KT is not enough, which verifies the superiority of the multi-relational framework.
- As illustrated, “ceof. MRT-KT” does not present improvements over the original transformer. This straightforward approach limits the adaptation for the model to learn different relations. On the other hand, “adapt. MRT-KT” shows a bit of enhancement suggesting that modeling different types of relations indeed helps improve the performance. The degeneration compared to MRT-KT indicates that the multi-relational attention mechanism provides more comprehensive modeling.

**5.2.3 Effect of Layer Number in Multi-Relational Self-Attention.** This part figures out how different layer numbers of multi-relational self-attention affect the final performance. We test it on the four datasets, and the results are shown in Figure 5. We find that one-layered multi-relational

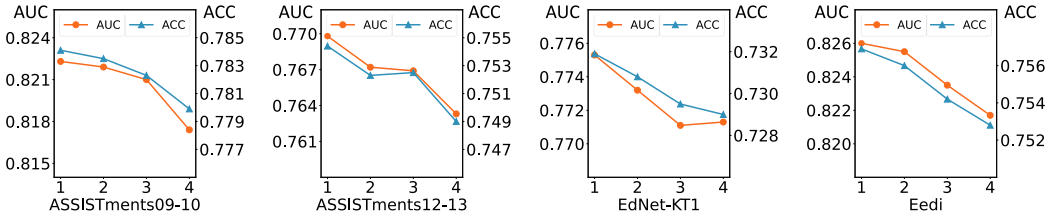


Fig. 5. Performance of multi-relational self-attention applying different numbers of layers on the four datasets.

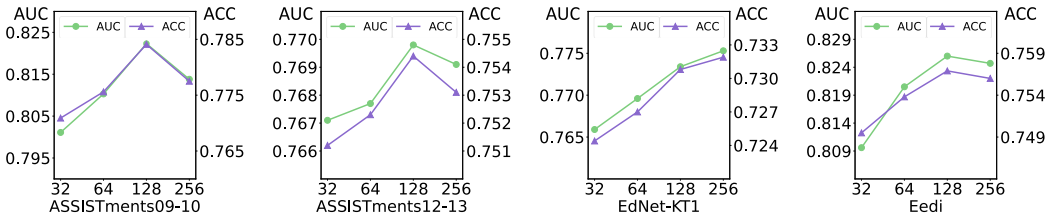


Fig. 6. Performance of MRT-KT w.r.t. different dimension sizes on the four datasets.

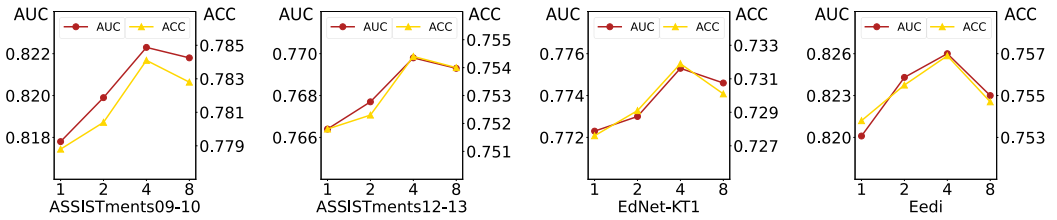


Fig. 7. Performance of MRT-KT w.r.t. different numbers of attention heads on the four datasets.

self-attention already achieves superior results and further increasing the layer number slightly lowers the performance. This shows that the two layers of interaction modeling (including the layer of target-dependent attention) are enough to capture mutual information within sequences, hence high-order interaction modeling does not introduce additional gains but might cause an overfitting issue.

**5.2.4 Effect of Hidden Dimension Size in MRT-KT.** This part figures out how different sizes of hidden dimensions affect the final performance. We test it on the four datasets as well, and the results are shown in Figure 6. We find that when the dimension size is small like 32 or 64, the performance is not very satisfactory, and the best dimension size is 128, except 256 for the EdNet-KT1 dataset. This demonstrates that fine-grained multi-relational interaction needs more complex modeling with more parameters. Meanwhile, the larger size of hidden dimensions may not fetch higher performance because of the overfitting problem.

**5.2.5 Effect of Attention Head Number in MRT-KT.** Previous studies [9, 28] have shown that the number of attention heads positively contributes to the final performance. In our multi-relational scheme, different numbers of heads can focus on different relations. Thus we present the analysis of how the number affects the final performance. The results on the four datasets are illustrated in Figure 7. We discover that the results on the first two datasets peak on the number 4 and decline at 8. This can be interpreted as the loss of balance when the network does not have enough heads

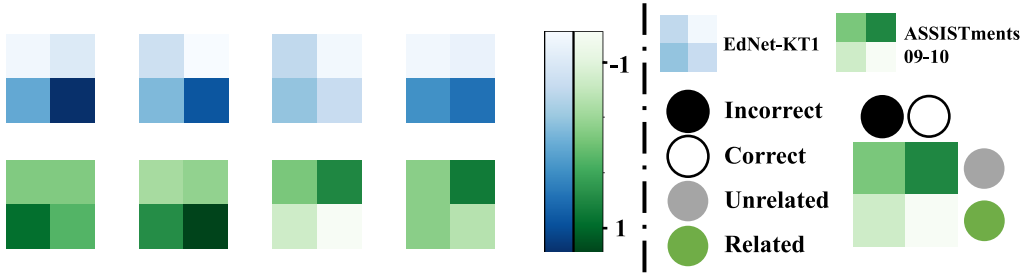


Fig. 8. Interaction weights visualization of four attention heads for both EdNet-KT1 and ASSISTments09–10.

to center on different relations. Furthermore, due to the reduction of hidden dimensions with the increase of heads, the modeling capability gets lower so the performance decreases.

**5.2.6 Interaction Visualization.** We visualize the normalized interaction weights (i.e.,  $a_{ij}^2$ ) in multi-relational target-dependent attention because they can reflect the interaction under specific response performance and knowledge concept sharing (or not). To make an intuitive comparison, for all datasets, we choose the model applying four attention heads. Figure 8 depicts the detailed interaction weights in different attention heads—four attention heads having four squares for a given dataset. Each square has four cells, the meaning of which is explained on the right side. The color of each cell is determined by the average interaction weights over the entire test set.

Based on the results, we observe that three attention heads deeply center on sharing concepts on EdNet-KT1, while on ASSISTments09–10, two heads focus on sharing concepts and the other two focus on not sharing concepts. This phenomenon conforms to the data analysis shown in Figure 2 that on EdNet-KT1, the cross effect between performance on two different questions is significantly larger if they share concepts. Furthermore, when sharing concepts, three of the heads pay more attention to correct responses on EdNet-KT1, but on ASSISTments09–10, three of the heads focus on incorrect responses. These discoveries are consistent with the results in the previous hyperparameter analysis—the effect of the attention head number. They both demonstrate that more attention heads empower the model with a higher capability to concentrate on multiple relation types.

**5.2.7 Temporal Kernel Analysis.** To validate the effectiveness of Relation-specific temporal kernel functions modeling the forgetting behavior of different relation types, we depict the curves in Figure 9. These curves indicate the temporal attention bias, which adjusts the effect of various relation types when predicting the target responses in the multi-relational target-dependent attention.

For all the datasets, the two *unrelated* effects show less impact than the *related* ones, especially for shorter time intervals on the two ASSISTments datasets, which adheres to the intuition. Considering all time effect, the *correct* ones present higher attention biases compared with the *incorrect* ones on the ASSISTments09–10, EdNet-KT1, and Eedi. From the time-sensitive aspect, *incorrect* lines skew more on both the ASSISTments09–10 and EdNet-KT1 datasets, which indicates that right answers could lead to better knowledge mastering. This phenomenon is not observed on ASSISTments12–13 where the *correct related* effect exhibits the most time sensitivity, and for Eedi the *incorrect related* skews more. On the whole, the results on the EdNet-KT1 and Eedi datasets show entirely different patterns from the others, especially when the time interval gets larger than 10 seconds (common in the real world), which is consistent with Figure 2 in Section 3.

**5.2.8 Case Study of Multi-Relational Target-Dependent Attention.** We visualize the attention weights of a student in Figure 10 to facilitate the understanding of how past question-response

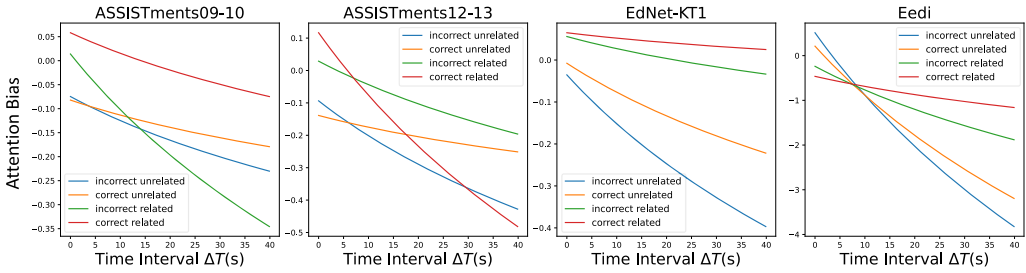


Fig. 9. Relation-specific temporal kernel functions of four relation types on four datasets in Multi-Relational Target-Dependent Attention. The *correct related* line refers to the kernel function of relation type *correct response and sharing same concepts*. Others could be understood in a similar way.

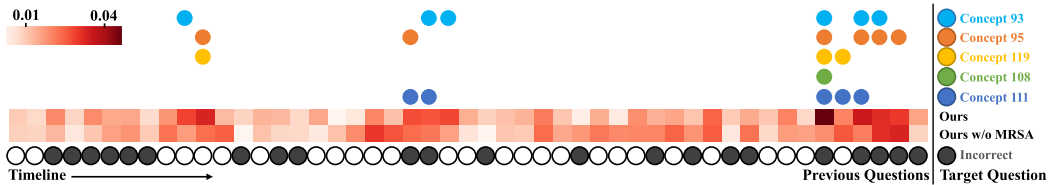


Fig. 10. Multi-relational target-dependent attention visualization of a student's (from EdNet-KT1) historical question-response sequence for a target question.

pairs affect KT prediction. The attention weights are averaged over four attention heads, and darker colors indicate larger weights. Moreover, the knowledge concepts of the target question are presented. And, if some of these concepts are associated with a past question, we plot the concepts on top of the question.

We compare the attention weights of MRT-KT and the variant that removes multi-relational self-attention, termed as Ours w/o MRSA. The results show the following findings. Firstly, both models reasonably give larger attention weights to the past questions that are relevant to the target question. In addition, with the increase of time step intervals, the attention weights of both models generally decay. This can be attributed to the relation-specific temporal kernels. Secondly, the model incorporating multi-relational self-attention further promotes attention concentration on the question with the same concepts, confirming the necessity of non-adjacent interaction modeling.

**5.2.9 Case Study of Multi-Relational Self-Attention.** To study the effect of modeling fine-grained non-adjacent interactions, we set the number of attention heads to 1 and visualize the normalized attention scores in the multi-relational self-attention component by giving two student cases of the two ASSISTments datasets in Figure 11. In a similar way, uncolored squares denote correctness and colored circles denote various concepts that are shown below the heatmaps to constitute the sequences with a length of 50. The timeline goes from left to right and darker colors indicate larger attention scores.

As can be seen, question-response pairs pay more attention to those sharing the same concepts, which indicates that the multi-relation encoding scheme considering concept relation works well. Another apparent discovery is that the model can clearly differentiate the property of correctness, e.g., focusing on past wrong answers more in these two cases. In addition, the attention scores on interactions themselves (i.e., self-relation) are larger on ASSISTments12–13. Therefore, the effectiveness of adding self-relation to the encoding scheme is verified. This also suggests that data records in ASSISTments12–13 have less sequentiality than in ASSISTments09–10.

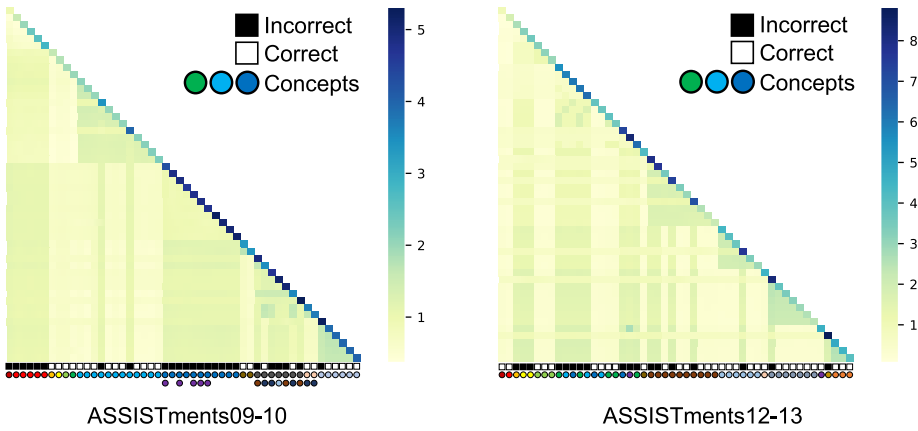


Fig. 11. Multi-relational self-attention visualization of two students' question-response sequences on the two ASSISTments datasets. To enhance the visibility for comparison, we multiply the attention values by their sequence length.

**5.2.10 Case Study of Knowledge Tracing.** For verifying the capability of MRT-KT to trace knowledge, we visualize the concept learning state of the same case in Section 5.2.9 and present the first half part for conciseness. Similarly, the uncolored circles are for correctness and the colored squares are for different concepts. The vertical arrows point to the time steps where we want to determine the mastering scores. To be specific, the timestamp we input the target concept is set at the moment when students finish answering the last question (i.e., timestamp plus response time in practice).

As illustrated in Figure 12, the *light blue* concept is learned in the middle of the sequence, where its score varies from 0.65 to 0.89. This indicates that correctly answering questions does enhance students' knowledge level. Meanwhile, the score decreases when this student gets an incorrect response (from 0.89 to 0.71). We also trace the scores of four main concepts at the end, where the *light blue* one is the highest. This can be attributed to the situation that questions related to the three other concepts are not well solved (i.e., no correct responses) by this student. Besides, comparing the *light blue* scores when the student has answered the last related question (0.85) and the last question (0.82), the forgetting behavior gets observed. This case study demonstrates that our method can model students' learning process and trace their knowledge states well.

## 6 APPLICATION

Besides enhancing prediction performance on offline datasets, MRT-KT can be also applied to real situations. Like other deep learning KT methods, a finely trained MRT-KT is able to be mounted on online tutoring systems to assist in serving students and educators. Its future response prediction can help students discriminate questions to practice and tutoring systems provide good suggestions. In addition, educators can trace each student's individual knowledge states to figure out which knowledge concept is well or poorly mastered, thus assigning personalized learning materials for them. Furthermore, MRT-KT displays a unique feature to capture historical responses' effects at a fine-grained level. Its calculated relation-specified attention for a student could help educators quantify contributions of his/her past responses, or different relations of question-responses, to correctly answer a new question. Moreover, the learned temporal kernels also provide insights for studying forgetting curves in different relation scenarios. One concern of MRT-KT to be grounded is efficiency. Its comprehensive fine-grained modeling introduces extra

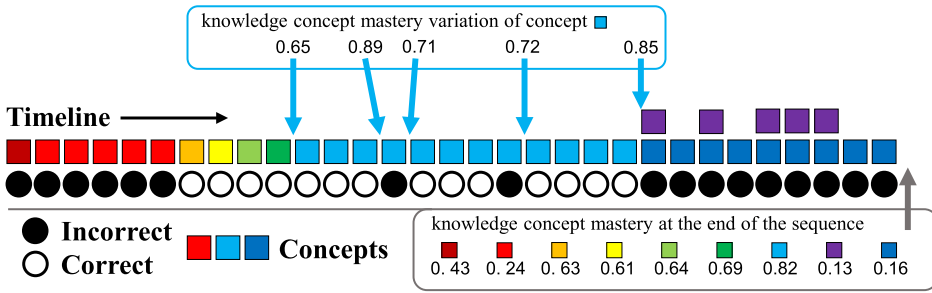


Fig. 12. Knowledge state tracing visualization of a student in ASSISTments09-10.

time complexity. But as illustrated in Section 4.7, the time consumption of MRT-KT is controlled in milliseconds so that it is completely acceptable for real applications.

## 7 CONCLUSION

In this work, we have studied the KT task of predicting student performance. Most of the existing methods have two limitations: (1) only considering the target-dependent interaction instead of the non-adjacent interaction and (2) ignoring the exploration of multiple relations within interactions. Afterward, we present data analysis and discover multiple types of correlations between question-response pairs. We then propose MRT-KT, a multi-relational transformer that encodes such multiple correlations into fine-grained interaction modeling to capture different cross effects. A relation encoding scheme is devised to determine a specific relation for each interaction based on knowledge concepts and student performance. In addition, relation-specific temporal kernels are presented to measure the forgetting behavior of multiple relations. Various detailed experiments are conducted to show that MRT-KT achieves superior performance by the fine-grained interaction modeling.

There still exists room for further improving this study. Like current mainstream KT approaches, the entire framework of MRT-KT is constructed on the traditional KT setting, i.e., predicting the binary response of each student. This setting is suitable for tracing knowledge when students solve choice and blank-filling questions. Nonetheless, another considerable part of questions is the open-ended questions (e.g., programming questions), which require students to demonstrate their solving steps. These questions could reflect the detailed knowledge mastery and provide more hints that why students can correctly answer the target questions. In the future, we will investigate to model the concrete problem solutions of students in MRT-KT and consider more fine-grained multi-relations (e.g., not limited to binary responses) among open-ended questions to capture more meaningful information.

## ACKNOWLEDGMENTS

The authors would like to thank the valuable comments of editors and reviewers.

## REFERENCES

- [1] Dejun Cai, Yuan Zhang, and Binta Dai. 2019. Learning path recommendation based on knowledge tracing model and reinforcement learning. In *Proceedings of the IEEE International Conference on Computer and Communications*. 1881–1885.
- [2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *Proceedings of the European Conference on Computer Vision*. 213–229.

- [3] Hao Cen, Kenneth Koedinger, and Brian Junker. 2006. Learning factors analysis—a general method for cognitive model evaluation and improvement. In *Proceedings of the International Conference on Intelligent Tutoring Systems*. 164–175.
- [4] Penghe Chen, Yu Lu, Vincent W. Zheng, and Yang Pian. 2018. Prerequisite-driven deep knowledge tracing. In *Proceedings of the IEEE International Conference on Data Mining*. 39–48.
- [5] Qiwei Chen, Huan Zhao, Wei Li, Pipei Huang, and Wenwu Ou. 2019. Behavior sequence transformer for e-commerce recommendation in alibaba. In *Proceedings of the Workshop on Deep Learning Practice for High-Dimensional Sparse Data*. 1–4.
- [6] Yuying Chen, Qi Liu, Zhenya Huang, Le Wu, Enhong Chen, Run-ze Wu, Yu Su, and Guoping Hu. 2017. Tracking knowledge proficiency of students with educational priors. In *Proceedings of the Conference on Information and Knowledge Management*. 989–998.
- [7] Song Cheng, Qi Liu, Enhong Chen, Kai Zhang, Zhenya Huang, Yu Yin, Xiaoqing Huang, and Yu Su. 2022. AdaptKT: A domain adaptable method for knowledge tracing. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*. 123–131.
- [8] Youngduck Choi, Youngnam Lee, Dongmin Shin, Junghyun Cho, Seoyon Park, Seewoo Lee, Jineon Baek, Chan Bae, Byungsoo Kim, and Jaewe Heo. 2020. Ednet: A large-scale hierarchical dataset in education. In *Proceedings of the International Conference on Artificial Intelligence in Education*. 69–73.
- [9] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? An analysis of BERT’s attention. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 276–286.
- [10] Albert T. Corbett and John R. Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction* 4 (1994), 253–278.
- [11] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc Viet Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 2978–2988.
- [12] Mostafa Dehghani, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Lukasz Kaiser. 2019. Universal transformers. In *Proceedings of the International Conference on Learning Representations*.
- [13] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 4171–4186.
- [14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkor, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proceedings of the International Conference on Learning Representations*.
- [15] Mingyu Feng, Neil Heffernan, and Kenneth Koedinger. 2009. Addressing the assessment challenge with an online system that tutors as it assesses. In *Proceedings of the User Modeling and User-Adapted Interaction*. 243–266.
- [16] Aritra Ghosh, Neil T. Heffernan, and Andrew S. Lan. 2020. Context-aware attentive knowledge tracing. In *Proceedings of the SIGKDD Conference on Knowledge Discovery and Data Mining*. 2330–2339.
- [17] Jibing Gong, Shen Wang, Jinlong Wang, Wenzheng Feng, Hao Peng, Jie Tang, and Philip S Yu. 2020. Attentional graph convolutional networks for knowledge concept recommendation in moocs in a heterogeneous view. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 79–88.
- [18] Zhenya Huang, Qi Liu, Yuying Chen, Le Wu, Keli Xiao, Enhong Chen, Haiping Ma, and Guoping Hu. 2020. Learning or forgetting? A dynamic approach for tracking the knowledge proficiency of students. *ACM Transactions on Information Systems* 38, 2, Article 19 (Feb. 2020), 33 pages.
- [19] Yujia Huo, Derek F. Wong, Lionel M. Ni, Lidia S. Chao, and Jing Zhang. 2020. Knowledge modeling via contextualized representations for LSTM-based personalized exercise recommendation. In *Proceedings of the Information Science*. 266–278.
- [20] Wang-Cheng Kang and Julian McAuley. 2018. Self-attentive sequential recommendation. In *Proceedings of the IEEE International Conference on Data Mining*. 197–206.
- [21] Tanja Käser, Severin Klingler, Alexander G. Schwing, and Markus H. Gross. 2017. Dynamic Bayesian networks for student modeling. In *Proceedings of the IEEE Transactions on Education*. 450–462.
- [22] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.
- [23] Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. 2019. Reformer: The efficient transformer. In *Proceedings of the International Conference on Learning Representations*.
- [24] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. In *Proceedings of the Nature*. 436–444.



- [25] Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Hui Xiong, Yu Su, and Guoping Hu. 2019. Ekt: Exercise-aware knowledge tracing for student performance prediction. In *Proceedings of the IEEE Transactions on Knowledge and Data Engineering*. 100–115.
- [26] Qi Liu, Shuanghong Shen, Zhenya Huang, Enhong Chen, and Yonghe Zheng. 2021. A survey of knowledge tracing. arXiv:2105.15106. Retrieved from <https://arxiv.org/abs/2105.15106>.
- [27] Frederic M. Lord. 2012. *Applications of Item Response Theory to Practical Testing Problems*. Routledge.
- [28] Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one?. In *Proceedings of the Conference on Neural Information Processing Systems*. 14014–14024.
- [29] Sein Minn, Yi Yu, Michel C. Desmarais, Feida Zhu, and Jill-Jënn Vie. 2018. Deep knowledge tracing and dynamic student classification for knowledge tracing. In *Proceedings of the IEEE International Conference on Data Mining*. 1182–1187.
- [30] Koki Nagatani, Qian Zhang, Masahiro Sato, Yan-Ying Chen, Francine Chen, and Tomoko Ohkuma. 2019. Augmenting knowledge tracing by considering forgetting behavior. In *Proceedings of the International World Wide Web Conference*. 3101–3107.
- [31] Shalini Pandey and George Karypis. 2019. A self attentive model for knowledge tracing. In *Proceedings of the Educational Data Mining*. 384–389.
- [32] Shalini Pandey and Jaideep Srivastava. 2020. RKT: Relation-aware self-attention for knowledge tracing. In *Proceedings of the Conference on Information and Knowledge Management*. 1205–1214.
- [33] Zachary A. Pardos and Neil T. Heffernan. 2010. Modeling individualization in a bayesian networks implementation of knowledge tracing. In *Proceedings of the ACM Conference on User Modeling, Adaptation and Personalization*. 255–266.
- [34] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J. Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. In *Proceedings of the Conference on Neural Information Processing Systems*. 505–513.
- [35] Shuanghong Shen, Qi Liu, Enhong Chen, Zhenya Huang, Wei Huang, Yu Yin, Yu Su, and Shijun Wang. 2021. Learning process-consistent knowledge tracing. In *Proceedings of the SIGKDD Conference on Knowledge Discovery and Data Mining*. 1452–1460.
- [36] Dongmin Shin, Yugeun Shim, Hangeul Yu, Seewoo Lee, Byungsoo Kim, and Youngduck Choi. 2021. Saint+: Integrating temporal features for ednet correctness prediction. In *Proceedings of the International Learning Analytics and Knowledge Conference*. 490–496.
- [37] Yu Su, Qingwen Liu, Qi Liu, Zhenya Huang, Yu Yin, Enhong Chen, Chris H. Q. Ding, Si Wei, and Guoping Hu. 2018. Exercise-enhanced sequential modeling for student performance prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*. 2435–2443.
- [38] Fei Sun, Jun Liu, Jian Wu, Changhua Pei, Xiao Lin, Wenwu Ou, and Peng Jiang. 2019. BERT4Rec: Sequential recommendation with bidirectional encoder representations from transformer. In *Proceedings of the Conference on Information and Knowledge Management*. 1441–1450.
- [39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the Conference on Neural Information Processing Systems*. 5998–6008.
- [40] Chenyang Wang, Weizhi Ma, Min Zhang, Chuancheng Lv, Fengyuan Wan, Huijie Lin, Taoran Tang, Yiqun Liu, and Shaoping Ma. 2021. Temporal cross-effects in knowledge tracing. In *Proceedings of the ACM International WSDM Conference*. 517–525.
- [41] Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*. 1705–1714.
- [42] Zichao Wang, Angus Lamb, Evgeny Saveliev, Pashmina Cameron, Jordan Zaykov, Jose Miguel Hernandez-Lobato, Richard E. Turner, Richard G. Baraniuk, Eedi Craig Barton, Simon Peyton Jones, Simon Woodhead, and Cheng Zhang. 2021. Results and Insights from Diagnostic Questions: The NeurIPS 2020 Education Challenge. In *Proceedings of the NeurIPS 2020 Competition and Demonstration Track (Proceedings of Machine Learning Research, Vol. 133)*, Hugo Jair Escalante and Katja Hofmann (Eds.). PMLR, 191–205.
- [43] Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*. 1112–1122.
- [44] Beverly Park Woolf. 2010. *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing e-learning*. Morgan Kaufmann.
- [45] Seongjun Yun, Minbyul Jeong, Raehyun Kim, Jaewoo Kang, and Hyunwoo J. Kim. 2019. Graph transformer networks. In *Proceedings of the Conference on Neural Information Processing Systems*. 11983–11993.

- [46] Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the International World Wide Web Conference*. 765–774.
- [47] Wei Zhang, Zeyuan Chen, Hongyuan Zha, and Jianyong Wang. 2021. Learning from substitutable and complementary relations for graph-based sequential product recommendation. In *Proceedings of the ACM Transactions on Information Systems*. 1–28.
- [48] Wayne Xin Zhao, Wenhui Zhang, Yulan He, Xing Xie, and Ji-Rong Wen. 2018. Automatically learning topics and difficulty levels of problems in online judge systems. *ACM Transactions on Information Systems* 36, 3, Article 27 (Mar. 2018), 33 pages.
- [49] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, and Li Zhang. 2021. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6881–6890.
- [50] Yuqiang Zhou, Qi Liu, Jinze Wu, Fei Wang, Zhenya Huang, Wei Tong, Hui Xiong, Enhong Chen, and Jianhui Ma. 2021. Modeling context-aware features for cognitive diagnosis in student learning. In *Proceedings of the SIGKDD Conference on Knowledge Discovery and Data Mining*. 2420–2428.

Received 16 May 2022; revised 5 January 2023; accepted 12 January 2023