Graph-Enhanced Multi-Activity Knowledge Tracing

Siqian Zhao(✉)[0009−0008−3913−7836] and Shaghayegh Sahebi[0000−0002−8933−3279]

Computer Science, University at Albany - SUNY, Albany, NY 12222, USA
{szhao2, ssahebi}@albany.edu

Abstract. Knowledge tracing (KT), or modeling student knowledge state given their past activity sequence, is one of the essential tasks in online education systems. Research has demonstrated that students benefit from both assessed (e.g., solving problems, which can be graded) and non-assessed learning activities (e.g., watching video lectures, which cannot be graded), and thus, modeling student knowledge from multiple types of activities with knowledge transfer between them is crucial. However, current approaches to multi-activity knowledge tracing cannot capture coarse-grained between-type associations and are primarily evaluated by predicting student performance on upcoming assessed activities (labeled data). Therefore, they are inadequate in incorporating signals from non-assessed activities (unlabeled data). We propose Graph-enhanced Multi-activity Knowledge Tracing (GMKT) that addresses these challenges by jointly learning a fine-grained recurrent memory-augmented student knowledge model and a coarse-grained graph neural network. In GMKT, we formulate multi-activity knowledge tracing as a semi-supervised sequence learning problem and optimize for accurate student performance and activity type at each time step. We demonstrate the effectiveness of our proposed model by experimenting on three real-world datasets.

Keywords: Educational data mining · Knowledge tracing · Knowledge transfer · Multi-activity · Transition-aware · Graph neural network

1 Introduction

The proliferation of large-scale online learning systems has facilitated distance education and provided students with access to a vast array of courses and diverse learning materials. One of the essential tasks in these systems is knowledge tracing (KT), which aims to model student knowledge based on their past interactions with the learning materials. Traditionally, KT models have focused on modeling assessed learning activities, such as solving problems and quizzes, and predicting students' performance in them [11,12,14,24,27]. However, recent research has recognized that students learn from both assessed and non-assessed learning activities, such as watching video lectures and studying worked examples [3,22]. Therefore, recently, multi-activity KT models [2,8,41,42] have

emerged to incorporate students' learning history of both assessed and non-assessed types of learning materials, resulting in more accurate predictions of students' future performance. However, these models still do not fully utilize the observations from non-assessed learning activities and cannot model long-range associations and complex knowledge transitions between learning materials.

More specifically, similar to their traditional counterparts, current multi-activity KT models are formulated as supervised sequence learning problems that predict students' future performance in non-assessed activities. Although these models incorporate non-assessed learning activities as input, they are not explicitly considered in the model's objective function, and therefore, they are not fully involved in optimization and training process. In effect, the non-assessed activities are underrepresented and their impact on student knowledge growth is diluted by these models. Moreover, similar to most modern KT models, multi-activity KTs are formulated as a form of recurrent neural network or tensor factorization models with Markovian assumptions that represent learning materials in fine-grained latent-concept spaces. Thus, the long-range and coarse-grained associations between learning materials are lost in these models. Furthermore, most multi-activity KTs represent all learning activity types in the same latent space and do not explicitly model student knowledge transfers when students transition between different activity types. These models overlook essential aspects of KT by ignoring processes by which student knowledge is attained, transferred, and materialized when transitions happen between various activity types.

To solve these challenges, we propose Graph-enhanced Multi-activity Knowledge Tracing (GMKT). GMKT fully represents both assessed and non-assessed learning activity and incorporates the complex, long-range associations among them. In GMKT, we represent the fine-grained learning material associations by developing a knowledge transfer layer, and the coarse-grained long-range associations by constructing a multi-activity graph neural network (GNN [15]) layer. We develop a transition-aware recurrent network for GMKT's knowledge transfer layer that traces student knowledge over different learning material types and learns knowledge transfer patterns among them using transition-specific knowledge transfer weight matrices. In GMKT's graph neural network layer, we construct a multi-activity transition graph according to the global transitions between learning materials and learn coarse-grained learning material representations by discovering transition-aware propagation and association matrices between them. Moreover, we formulate multi-activity KT as a semi-supervised learning problem and introduce a new activity-type learning objective for GMKT that uses the student's choice of learning activity type as an additional signal in training the model. To summarize, the main contributions of this work are:

- We propose two transition-aware multi-activity recurrent and graph neural networks in GMKT that jointly represent fine-grained and long-range coarse-grained associations between different types of learning materials.
- We formulate the multi-activity knowledge tracing, with a novel perspective, as a semi-supervised sequence learning task and add an activity type objective to GMKT's optimization problem to fully discern the signals from non-assessed

learning activities.
• We demonstrate the effectiveness of GMKT on three real-world datasets by comparing it with 15 baseline methods from various research lines, conducting ablation studies, and performing sensitivity analysis.
• We showcase the efficacy of GMKT's transition-aware knowledge transfer by analyzing knowledge transfer weight matrices between different material types.

2 Related Work

Knowledge Tracing: KT approaches mainly rely on the predefined association between learning material and knowledge concepts or components [5,11,13,26], such as BKT [11] and Regression-based KT methods [5,6,13,18,26]. These approaches measure student knowledge of learning material by quantifying student mastery level of the set of knowledge concepts [13,18,19,21]. Later, models like DKT and DKVMN have been proposed to learn the underlying latent concepts of the learning materials [14,17,24,27,30,32,39,39], since predefined mapping between materials and concepts is typically labeled by human experts, which is costly and impractical for nowadays large-scale online education systems.

All these methods focus on assessed learning materials and do not model students' non-assessed learning activities. Zhang et al. and Choi et al. suggest including non-assessed activities as additional features in modeling student knowledge [8,40]. However, these models do not explicitly measure a student's knowledge state when interacting with non-assessed materials. To the best of our knowledge, there are only a few multi-activity KT approaches that explicitly model student knowledge from multi-type students' learning activities, including MA-Elo [2], MA-FM [1], MVKM [41], DMKT [34], TAMKOT [42]. However, except TAMKOT, these methods either require a predefined mapping between the learning materials and concepts, or explicitly represent the dynamics of knowledge transfer among different learning activities. Moreover, including TAMKOT, none of the methods mentioned above consider the global neighborhood-based transitions and have an activity-type learning objective.
Graph Neural Network: More recently, GNN [15] is widely used to learn and represent the structural information of a graph. It has been shown success in various domains [28,29,29,35,35,37]. Existing GNN-based KT methods include GKT [23], GIKT [36], PEBG [20], SKT [33], and DGEKT [12]. Except for DGEKT building graphs through learning activities, these GNN-based methods all create graphs between learning materials or knowledge concepts, neglecting the global transition-structured information from student activity sequences. Additionally, all of the previous GNN-based methods focus on single-type learning material, while we propose to build graphs for multi-type materials.

3 Problem Formulation

In this work, our goal is to model and trace student knowledge by practicing both assessed and non-assessed learning activities. Assuming that there are two

types of learning materials, one assessed (e.g., questions) and one non-assessed (e.g., video lectures), we represent a student's whole trajectory of activities as a sequence of tuples, $\{\langle i_1, d_1\rangle, ..., \langle i_t, d_t\rangle\}$, where each tuple $\langle i_t, d_t\rangle$ indicates a student's activity at time step t. Here, $d_t \in \{0, 1\}$ is a binary indicator that represents the learning activity type at time step t, with 0 denoting assessed and 1 denoting non-assessed type, and i_t indicates the learning material being interacted with. Specifically, we formulate i_t as: $i_t = \begin{cases} (q_t, r_t) & \text{if } d_t = 0 \\ l_t & \text{if } d_t = 1 \end{cases}$, where (q_t, r_t) represents the student's interaction with the assessed material q_t at time step t, with performance r_t, and l_t represents the non-assessed material that the student interacted with at time step t. Conventionally, knowledge tracing is evaluated by the task of performance prediction in the target student's upcoming assessed learning activity q_{t+1}, based on their past assessed activity records $\{(q_1, r_1), \ldots, (q_t, r_t)\}$. Here, given a student's past assessed and non-assessed learning activity history, $\{\langle i_1, d_1\rangle, \ldots, \langle i_t, d_t\rangle\}$, we aim to predict their upcoming performance on the assessed material q_{t+1} at time step $t + 1$.

4 Graph-Enhanced Multi-Activity Knowledge Tracing

Our model, Graph-enhanced Multi-activity Knowledge Tracing (GMKT), comprises four key layers, including (1) The embedding layer for encoding each student activity into a latent concept feature space; (2) The multi-activity transition graph layer that incorporates the coarse-grained long-range patterns among learning materials; (3) The recurrent knowledge transfer layer that captures student knowledge and fine-grained transfers as students transition between different activities; and (4) The prediction layer that generates a prediction of a student's upcoming performance on an assessed material. We introduce the details of each layer in the next sections and show GMKT's architecture in Figure 1.

Notations. We use lowercase letters, boldface lowercase letters, and boldface capital letters to respectively denote scalars (q_t), vectors($\boldsymbol{q_t}$), and matrices ($\boldsymbol{A}^q$).

4.1 Embedding Layer

The embedding layer is designed to learn the embedding of each learning activity i_t, which is then used as input for capturing the students' knowledge state and transfer from the latent concept space. To do this, GMKT learns the latent representation of the material (q_t and l_t) and the student response (r_t) for activity i_t. Assuming two learning material types, questions and video lectures, we embed each material type separately. This design allows for a more flexible representation by allowing different embedding sizes for each material type. Specifically, GMKT learns two underlying latent embedding matrices $\boldsymbol{A}^q \in \mathbb{R}^{N^Q \times d_q}$ and $\boldsymbol{A}^l \in \mathbb{R}^{N^L \times d_l}$ to respectively map all questions and lectures to their specified latent spaces. Here, N^Q and N^L are the number of questions and video lectures, and d_q and d_l are the respective latent embedding sizes. To incorporate student performance outcomes in assessed activities, GMKT maps r_t into a

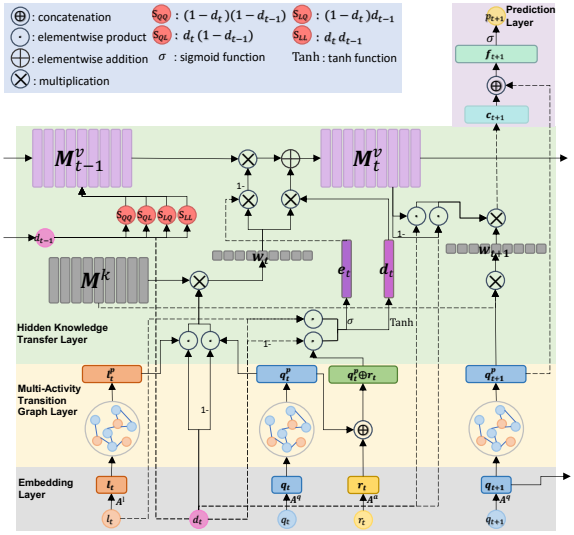

Fig. 1: The architecture of the GMKT model. The solid and dashed lines are identical. Different line types are used to clarify between lines that cross/fall over each other.

higher-dimensional performance latent space. We consider two scenarios for r_t, namely, binary outcomes (e.g., correctness in solving a question) and numerical outcomes (e.g., normalized exam scores between 0 and 1). For the binary case, we learn an embedding matrix $\boldsymbol{A}^r \in \mathbb{R}^{2\times d_r}$ to map r_t, where d_r is the performance embedding size. For the numerical case, we use $\boldsymbol{A}^r \in \mathbb{R}^{d_r}$, and apply a linear mapping function $f(r_t) = r_t\boldsymbol{A}^r$ to the performance r_t.

4.2 Multi-Activity Transition Graph Layer

Student learning activity sequences can provide coarse-grained insights into relationships between different learning materials. Observing students interacting with materials consecutively may indicate that they are similar or related. To capture such coarse-grained aggregate information, we construct a multi-activity transition graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ consists of all assessed and non-assessed learning materials as nodes, and $\mathcal{E}$ represents the undirected edges between materials that correspond to transitions between materials in a student's sequence. An edge exists between two materials if a student from the training sessions has interacted with them consecutively. For example, given a student's sequence $\{\langle(question_1, 0), 0\rangle, \langle lecture_4, 1\rangle, \langle(question_2, 1), 0\rangle, ...\}$, edges between $question_1$ and $lecture_4$, as well as $question_2$ and $lecture_4$, are added to graph.

To update a learning material's representation, we use propagation matrices to integrate the embedding of that learning material with its neighboring materials. Having assessed and non-assessed types of learning materials and their different contributions, we also learn transition matrices to map the two types

to each other. Specifically, taking the material's embedding $\boldsymbol{q}_t$ or $\boldsymbol{l}_t$ from the embedding layer as the input, the material aggregation is formulated as:

$$\boldsymbol{q}_t^p = \boldsymbol{V}_Q^\intercal \left[\boldsymbol{q}_t + \frac{1}{|\mathcal{N}_{q_t}^Q|} \sum_{i \in \mathcal{N}_{q_t}^Q} \boldsymbol{G}_{QQ}^\intercal \boldsymbol{q}_i + \frac{1}{|\mathcal{N}_{q_t}^L|} \sum_{j \in \mathcal{N}_{q_t}^L} \boldsymbol{G}_{QL}^\intercal \boldsymbol{l}_j \right] + \boldsymbol{b}_Q \tag{1}$$

$$\boldsymbol{l}_t^p = \boldsymbol{V}_L^\intercal \left[\boldsymbol{l}_t + \frac{1}{|\mathcal{N}_{l_t}^L|} \sum_{i \in \mathcal{N}_{l_t}^L} \boldsymbol{G}_{LL}^\intercal \boldsymbol{l}_i + \frac{1}{|\mathcal{N}_{l_t}^Q|} \sum_{ij \in \mathcal{N}_{l_t}^Q} \boldsymbol{G}_{LQ}^\intercal \boldsymbol{q}_j \right] + \boldsymbol{b}_L \tag{2}$$

where $\boldsymbol{q}_t^p$ and $\boldsymbol{l}_t^p$ represent the coarse-grained embeddings of learning material q_t and l_t after the GNN propagation. Transition matrices $\boldsymbol{G}_{QQ} \in \mathbb{R}^{d_q \times d_q}$, $\boldsymbol{G}_{QL} \in \mathbb{R}^{d_l \times d_q}$, $\boldsymbol{G}_{LL} \in \mathbb{R}^{d_l \times d_l}$, and $\boldsymbol{G}_{LQ} \in \mathbb{R}^{d_q \times d_l}$ are learned to map each material type's embeddings to corresponding material space for propagation. $\mathcal{N}_{**}^*$ denotes the set of neighbors from type * for the material **. For example, $\mathcal{N}_{q_t}^L$ denotes all the lecture neighbors ("L") of question ("Q") q_t. $\boldsymbol{V}_Q^\intercal \in \mathbb{R}^{d_q \times d_q}$ and $\boldsymbol{V}_L^\intercal \in \mathbb{R}^{d_l \times d_l}$ are weight matrices for propagation, $\boldsymbol{b}_Q \in \mathbb{R}^{d_q}$ and $\boldsymbol{b}_L \in \mathbb{R}^{d_l}$ are bias terms.

In this layer, in addition to the coarse-grained associations, the neighborhood-based propagation enables the discovery of long-range relationships between materials that cannot be easily captured in the recurrent knowledge transfer layer of the architecture.

4.3 Knowledge Transfer Layer

We design the knowledge transfer layer to accurately learn the dynamic student knowledge state and the fine-grained material representations. To do so, similar to dynamic key-value memory networks (DKVMN) [39], we employ a static key matrix $\boldsymbol{M}^k \in \mathbb{R}^{N \times d_k}$ to represent N latent concept features and a dynamic value matrix $\boldsymbol{M}_t^v \in \mathbb{R}^{N \times d_v}$ to track the student's mastery state in them. Each vector in the static key matrix corresponds to a concept characterized by d_k latent concept features, while each vector in the dynamic value matrix is a d_v-size memory slot to monitor the student's updated knowledge state (mastery levels) of the corresponding concept over time steps.

Unlike DKVMN, GMKT further models different activity types and the transitions among them. As the way knowledge transfers between different material types can vary depending on the order of the transition, we learn a unique knowledge transfer pattern for each transition between every two distinct material types. To model these transition-specific transfer patterns, we incorporate current and previous activity types as additional inputs. GMKT uses a set of indicators to activate corresponding knowledge transfer weight at each time t. Having two material types, questions ("Q") and lectures ("L"), four transition indicators at each time t are formulated based on material types d_t and d_{t-1}:

$$s_{QQ} = (1 - d_t)(1 - d_{t-1}) \quad s_{QL} = d_t(1 - d_{t-1}) \quad s_{LQ} = (1 - d_t)d_{t-1} \quad s_{LL} = d_t d_{t-1} \tag{3}$$

At each time step t, only one of the above transition indicators is equal to 1, while the rest are 0. For example, $s_{QL} = 1$ and $s_{QQ} = s_{LQ} = s_{LL} = 0$ indicate that the student has transitioned from attempting a question at time $t-1$

to watching a video lecture at time t. Then, the transition indicators s_{**} are utilized to activate the corresponding transition-specific weight matrices $\boldsymbol{T}_{**}$ for updating the student's knowledge state $\boldsymbol{M}_t^v$. Consequently, GMKT first computes the attention weight vector $\boldsymbol{w}_t$, which represent the correlation between learning material (q_t or l_t) and each of the N latent concepts. The coarse-grained embedding of the material ($\boldsymbol{q}_t^p$ or $\boldsymbol{l}_t^p$) from equation 1 and 2, and the static key matrix $\boldsymbol{M}^k$ are used to compute $\boldsymbol{w}_t \in \mathbb{R}^N$ as follows:

$$w_t(i) = softmax([(1-d_t)\cdot \boldsymbol{R}_q^\top \boldsymbol{q}_t^p + d_t \cdot \boldsymbol{R}_l^\top \boldsymbol{l}_t^p]^\top \boldsymbol{M}^k(i)) \tag{4}$$

where $w_t(i)$ is the i-th element in the attention weight vector $\boldsymbol{w}_t$, and the Softmax function $softmax(m_i) = e^{m_i}/\sum_j e^{m_j}$ is to ensure that the attention weights sum to one. $\boldsymbol{R}_q \in \mathbb{R}^{d_q \times d_k}$ and $\boldsymbol{R}_l \in \mathbb{R}^{d_l \times d_k}$ are used to map question and lecture activity embedding to the concept feature space of $\boldsymbol{M}^k$ in size d_k.

Then, at each time step t, the student's knowledge state is updated based on the learning activity i_t ((q_r, r_t) or l_t), using the *erase-followed-by-add* mechanism to modify the memory value matrix $\boldsymbol{M}_t^v$. It involves erasing previous redundant information before adding new information to $\boldsymbol{M}_t^v$ and is formulated as follows:

Erase:

$$\boldsymbol{e}_t = \sigma\big((1-d_t)\cdot \boldsymbol{E}_q^\top[\boldsymbol{q}_t^p \oplus \boldsymbol{r}_t] + d_t \cdot \boldsymbol{E}_l^\top \boldsymbol{l}_t^p + \boldsymbol{b}_e\big) \tag{5}$$

$$\begin{aligned}\tilde{\boldsymbol{M}}_t^v(i) = &[s_{QQ}\cdot \boldsymbol{T}_{QQ}\boldsymbol{M}_{t-1}^v + s_{LL}\cdot \boldsymbol{T}_{LL}\boldsymbol{M}_{t-1}^v \\ &+ s_{QL}\cdot \boldsymbol{T}_{QL}\boldsymbol{M}_{t-1}^v + s_{LQ}\cdot \boldsymbol{T}_{LQ}\boldsymbol{M}_{t-1}^v](i)\cdot[\boldsymbol{1} - w_t(i)\boldsymbol{e}_t]\end{aligned} \tag{6}$$

Add:

$$\boldsymbol{d}_t = Tanh\big((1-d_t)\cdot \boldsymbol{D}_q^\top[\boldsymbol{q}_t^p \oplus \boldsymbol{r}_t] + d_t \cdot \boldsymbol{D}_l^\top \boldsymbol{l}_t^p + \boldsymbol{b}_d\big) \tag{7}$$

$$\boldsymbol{M}_t^v(i) = \tilde{\boldsymbol{M}}_t^v(i) + w_t(i)\boldsymbol{d}_t \tag{8}$$

Here, σ and $Tanh$ are Sigmoid and Tanh activation functions. The erase vector $\boldsymbol{e}_t \in [0,1]^{d_v}$ is formulated to remove redundant knowledge information from $\boldsymbol{M}_{t-1}^v$. The add vector $\boldsymbol{d}_t \in \mathbb{R}^{d_v}$ is formulated to capture the new knowledge that the student acquires at time t. $\tilde{\boldsymbol{M}}_t^v(i)$ and $\boldsymbol{M}_t^v(i)$ indicates the i-th knowledge slot of $\boldsymbol{M}_t^v$ after erasing and adding process. We acknowledge that knowledge transfer can differ for the four possible transitions among different learning material types, therefore, separate transfer weight matrices are utilized. These matrices are activated by using the four different transition indicators s_{**}, namely $\boldsymbol{T}_{QQ}$, $\boldsymbol{T}_{QL}$, $\boldsymbol{T}_{LQ}$, and $\boldsymbol{T}_{LL} \in \mathbb{R}^{d_v \times d_v}$. For example, when the student switches from watching video lectures to solving questions, $\boldsymbol{T}_{LQ}$ represents knowledge transfer from the previous student knowledge state $\boldsymbol{M}_{t-1}^v$ to the current state and it is activated since $s_{LQ} = 1$. In addition, $(1-d_t)$ and (d_t) are used to determine whether the learning activity i_t is a question or a lecture attempt. They are used to activate the corresponding matrices $\boldsymbol{E}_q$ and $\boldsymbol{D}_q \in \mathbb{R}^{(d_q+d_r)\times d_v}$, $\boldsymbol{E}_l$ and $\boldsymbol{D}_l \in \mathbb{R}^{d_l \times d_v}$ for mapping the learning activity embedding to concept feature space of value matrix. $\boldsymbol{b}_e$ and $\boldsymbol{b}_d \in \mathbb{R}^{d_v}$ represent the bias terms.

In this layer, representing student knowledge and learning material concepts in fine-grained latent features and the transition-aware transfer matrices allow for more precise student performance prediction and capture more detailed associations between consequent learning materials in a sequence.

4.4 Prediction Layer

In this layer, GMKT predicts the performance of a student on a given question q_{t+1} at the next time $t+1$, based on their knowledge state of the q_{t+1}'s concepts.

$$w_{t+1}(i) = softmax([\boldsymbol{R}_q^\top \boldsymbol{q}_{t+1}^p]^\top \boldsymbol{M}^k(i)) \tag{9}$$

$$\boldsymbol{c}_{t+1} = \sum_{i=1}^{N} w_{t+1}(i)\big[(1-d_t)\cdot \boldsymbol{M}_t^v \boldsymbol{T}_{QQ} + d_t \cdot \boldsymbol{M}_t^v \boldsymbol{T}_{LQ}\big](i) \tag{10}$$

$$\boldsymbol{f}_{t+1} = Tanh(\boldsymbol{W}_f^\top [\boldsymbol{c}_{t+1} \oplus \boldsymbol{q}_{t+1}] + \boldsymbol{b}_f) \tag{11}$$

Initially, the correlation between question q_{t+1} and each of the N latent concepts is determined by computing the attention weight vector $\boldsymbol{w}_{t+1}$ (equation 9). The read content $\boldsymbol{c}_{t+1}$ is then retrieved to summarize the student's knowledge state of question q_{t+1} by using the weighted sum of all memory slots in the value matrix $\boldsymbol{M}_t^v$ and $\boldsymbol{w}_{t+1}$ (equation 10). Here, $(1-d_t)$ and d_t are used to indicate whether the knowledge transfer from time t to $t+1$ for predicting the performance of q_{t+1} is from a question or a lecture. Next, the concatenation of $\boldsymbol{c}_{t+1}$ and the next question's embedding vector $\boldsymbol{q}_{t+1}$, is passed through a fully connected layer with a Tanh activation function to obtain a summary vector $\boldsymbol{f}_{t+1}$(equation 11), where $\boldsymbol{W}_f \in \mathbb{R}^{(d_v+d_q)\times d_s}$ and $\boldsymbol{b}_f \in \mathbb{R}^{d_s}$ is the weight matrix and the bias term, with d_s is the summary vector size. Finally, another fully connected layer with the Sigmoid activation function is used upon $\boldsymbol{f}_{t+1}$ to predict the student's performance p_{t+1}:

$$p_{t+1} = \sigma(\boldsymbol{W}_p^\top \boldsymbol{f}_{t+1} + b_p) \tag{12}$$

where a scalar p_{t+1} represents the probability of the student correctly answering the next question q_{t+1}, $\boldsymbol{W}_p \in \mathbb{R}^{d_s \times 1}$ and $b_p \in \mathbb{R}$ are weight matrix and bias term.

4.5 Optimization and Objective Function

Similar to traditional KT models, we aim to minimize the following binary cross-entropy loss between actual and estimated student performance r_t and p_t:

$$\mathcal{L} = -\sum_t (r_t \log p_t + (1-r_t)\log(1-p_t)) \tag{13}$$

But, unlike previous KT models, our goal is to also learn from the unlabeled data (non-assessed activities). To do so, we propose an additional objective to accurately estimate the type of the next material. Accordingly, we propose a read content of learning material type $\boldsymbol{c}_t^o$ to summarize a student's behavior state of material type at each time t by using an attention weight vector, denoted by $\boldsymbol{w}_t^o$:

$$w_t^o(i) = softmax(\big[(1-d_t)\cdot \boldsymbol{O}_q^\top \boldsymbol{q}_t^p + d_t \cdot \boldsymbol{O}_l^\top \boldsymbol{l}_t^p\big]^\top \boldsymbol{M}^k(i)) \tag{14}$$

$$\boldsymbol{c}_t^o = \sum_{i=1}^{N} w_t^o(i) \boldsymbol{M}_t^v(i) \tag{15}$$

where $w_t^o(i)$ is the i-th element of $\boldsymbol{w}_t^o$, and $\boldsymbol{O}_q \in \mathbb{R}^{d_q \times d_k}$ and $\boldsymbol{O}_l \in \mathbb{R}^{d_l \times d_k}$ and two weight matrices to map question and lecture embeddings. We then model the type of material the student will interact with at time $t+1$ using equation 16:

$$\boldsymbol{p}_{t+1}^o = \sigma(d_t \cdot \boldsymbol{W}_{oq}^T \boldsymbol{c}_t^o + (1-d_t)\cdot \boldsymbol{W}_{ol}^T \boldsymbol{c}_t^o + b_o) \tag{16}$$

Table 1: Descriptive statistics of datasets.

Dataset	#Students	#Assessed Materials	#Assessed Activities	Assessed Responses Mean	Assessed Responses STD	#Correct Assessed Responses	#Incorrect Assessed Responses	#Non-assessed Materials	#Non-assessed Activities
EdNet	1000	11249	200931	0.5910	0.2417	118747	82184	8324	150821
Junyi	2063	3760	290754	0.6660	0.2224	193664	97090	1432	69050
MORF	686	10	12031	0.7763	0.2507	N/A	N/A	52	41980

where p^o_{t+1} represents the probability that the next learning material student will interact be a question. $\boldsymbol{W}_{oq}$ and $\boldsymbol{W}_{ol} \in \mathbb{R}^{d_v \times 1}$ are two weight matrices, $b_o \in \mathbb{R}$ is the bias term. Finally, the activity-type objective function $\mathcal{L}^o$ is formulated as a binary cross-entropy loss between p^o_t and the actual material type d_t:

$$\mathcal{L}^o = -\sum_t \left(d_t \log p^o_t + (1 - d_t) \log \left(1 - p^o_t\right)\right) \tag{17}$$

Eventually, we minimize a combination of the performance objective function $\mathcal{L}$ Equation 13) and the activity-type objective function $\mathcal{L}^o$ (equation 17) with a regularization term to learn the parameters of GMKT, as shown in equation 18:

$$\mathcal{L}_{total} = \mathcal{L} + \lambda_o \mathcal{L}^o + \lambda_\theta ||\theta||^2 \tag{18}$$

We use λ_o to balance between the contribution of student performance objective and activity-type objective. θ represents the set of all trainable parameters in GMKT, and the term $||\theta||^2$ corresponds to the regularization, while λ_θ denotes the hyperparameter that determines the weight of this regularization term.

5 Experiments

We evaluate GMKT through two sets of experiments. First, we compare GMKT's student performance predictive ability with baseline KT methods and perform ablation studies and sensitivity analysis of the model's components. Then, we compare transition weight matrices to examine knowledge transfer between learning material types. Our code and supplementary material are available on GitHub [1].

5.1 Datasets

We use three real-world datasets for our experiments. Table 1 provides an overview of the general statistics for each dataset.

EdNet [2] [9]: This dataset is collected from Santa [3], a multi-platform AI tutoring service that was designed to provide Korean students with a platform to practice for TOEIC [4] English testing. Every time, students choose a bundle that includes a set of problems to practice, and optional corresponding problem explanations to read. We use the preprocessed data introduced in [42] for our experiments,

[1] https://github.com/persai-lab/2023-ECML-PKDD-GMKT
[2] https://github.com/riiid/ednet
[3] https://www.aitutorsanta.com/
[4] https://www.ets.org/toeic

which use problems (assessed) and their associated problem explanations (non-assessed) as two types of learning materials.

Junyi[5] [10]: This dataset is sourced from a Chinese e-learning website that teaches math to students. The website covers eight math areas with varying difficulty levels. For our experiments, we use the preprocessed data made available in [7,42], with problems (assessed) and hints (non-assessed) as two distinct learning material types. Each problem may be associated with multiple hints. During practice, students have the option to request hints for solving problems.

MORF [4]: This dataset comprises data from an online course "Educational Data Mining" offered on Coursera[6] and accessed from the MOOC Replication Framework (MORF) platform[7]. The course consists of modules covering various topics, such as "classification". During the course, students are expected to watch several video lectures per module and complete an assignment, containing multiple problems. However, only coarse-grained assignment-level data is available. Thus, we treat each submission of an assignment as one assessed activity and consider the overall score as the activity response. For our experiments, the two material types are assignments (assessed) and video lectures (non-assessed).

5.2 Baselines

To evaluate our proposed method on student performance prediction task, we compare it with six state-of-the-art assessed-only supervised KT models and three multi-activity KT models. In addition, to ensure a fair comparison, we also extend the six assessed-only supervised KT models to handle both assessed and non-assessed activities and also include a multi-layer perceptron (MLP) baseline that can handle both types of activities. We denoted these extended models by "original model name +M". Overall, we evaluated our method against 15 baselines, consisting of eight deep learning-based models and one tensor factorization model among the original nine baselines. Notably, to ensure fairness, we refrain from comparing with GNN-based KT models mentioned in section 2, as they require the predefined mapping between materials and concepts, whereas we learn the underlying latent concept. For baselines that originally used the knowledge concept of each question as inputs (e.g., DKT), we used each question as a knowledge component. The assessed supervised KT baselines are:

DKT [27] employs recurrent neural networks to model the knowledge state of students, and is the first deep learning-based KT method.

DKVMN [39] modifies MANN that utilizes a static key matrix to represent knowledge concepts and a dynamic value matrix to update student knowledge.

DeepIRT [38] extends DKVMN by incorporating the one-parameter logistic item response theory, which provides better interpretability of KT.

SAKT [24] applies a self-attentive mechanism to model the inter-dependencies between student interactions and improve the effectiveness of KT.

[5] https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=1275

[6] https://www.coursera.org/

[7] https://educational-technology-collective.github.io/morf/

SAINT [8] is a transformer-based method and is an encoder-decoder model that employs deep self-attentive layers to separately encode exercises and responses.
AKT [14] is a context-aware KT model that utilizes a monotonic attention mechanism to summarize the impact of past student activity performance on the current activity's knowledge state.
The baseline methods support both assessed and non-assessed activities are:
DKT+M [40], **DKVMN+M**, **SAINT+M** [8], **AKT+M** and **AKT+M** are variants of DKT, DKVMN, SAINT, SAKT, and AKT. in these extended models, non-assessed learning activities embedding are summarized as an additional feature, with the problem embedding as the model input.
MLP+M [16] is a simple multi-layer perceptron that takes the embedding of a student's three most recent assessed activities and three non-assessed activities as input to predict student knowledge of a concept.
MVKM [41] can model student knowledge acquisition from multi-type learning activities. It is a method based on multi-view tensor factorization that constructs separate tensors for student activities from each learning material type but cannot explicitly capture the knowledge transition between material types.
DMKT [34] is based on DKVMN and models distinct read and write operations for assessed and non-assessed learning material types. However, it lacks the ability to explicitly model knowledge transfer between assessed and non-assessed learning materials. Moreover, it requires a fixed number of non-assessed learning activities between every two assessed ones, making it less flexible in modeling the student knowledge from the complete activity sequence.
TAMKOT [42] is a transition-aware KT model that builds based on LSTM. It learns multiple knowledge transfer matrices to explicitly model the knowledge transfer between different activity types. However, it does not consider the global neighborhood-based transitions its knowledge modeling layer is LSTM-based, and its objective function only considers students' assessed activities.

5.3 Experiment Setup

We adopt 5-fold student stratified cross-validation, following standard KT experiments [27,34]. In each fold, 80% of students' sequences are randomly chosen as the training set, while the remaining 20% of students' sequences are used as the test set. For hyperparameter tuning, we separate 20% of students from training set and use their sequences as the validation set. We conduct a coarse-grained grid search to find the best hyperparameters, which are reported in Table 2.

5.4 Student Performance Prediction

In student performance prediction experiments, we report the mean results across five folds of each method and present the paired t-test p-values that compare each baseline to GMKT. For datasets where student performance is binary (correctness), such as EdNet and Junyi, we evaluate model performance using Area Under Curve (AUC). For datasets where student performance is numeric values (scores), such as MORF, we normalize student assignment scores within

Table 2: Best learned hyperparameters

Dataset	d_q	d_r	d_l	d_k	d_v	d_s	N	λ_o	λ_θ
EdNet	64	32	32	32	32	32	8	0.1	0.05
Junyi	32	32	32	64	64	32	32	0.1	0.05
MORF	32	16	8	32	32	32	8	0.05	0.03

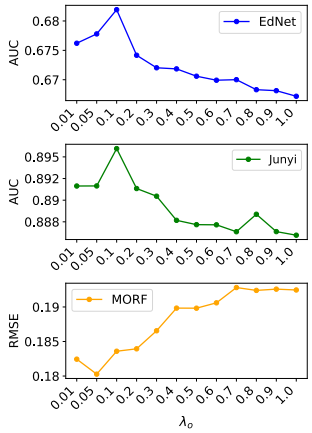

Fig. 2: Performance w.r.t. λ_o

Table 3: Student performance prediction results. The best and second-best results are in bold and underlined. $**$ and $*$ represent paired t-test $p-values < 0.05$ and < 0.1, compared to GMKT.

Methods	EdNet AUC	Junyi AUC	MORF RMSE
DKT	0.6393^{**}	0.8623^{**}	0.1990^{**}
DKVMN	0.6296^{**}	0.8558^{**}	0.1995^{**}
SAKT	0.6334^{**}	0.8053^{**}	0.1975^{**}
SAINT	0.5205^{**}	0.7951^{**}	0.2190^{**}
AKT	0.6393^{**}	0.8093^{**}	0.2417^{**}
DeepIRT	0.6290^{**}	0.8498^{**}	0.1946^{**}
DKT+M	0.6372^{**}	0.8652^{*}	0.1942^{**}
DKVMN+M	0.6343^{**}	0.8513^{**}	0.2071^{**}
SAKT+M	0.6323^{**}	0.7911^{**}	0.1981^{**}
SAINT+M	0.5491^{**}	0.7741^{**}	0.2007^{**}
AKT+M	0.6404^{**}	0.8099^{**}	0.2226^{**}
MLP+M	0.6102^{**}	0.7290^{**}	0.2428^{**}
MVKM	–	–	0.1936^{*}
DMKT	0.6394^{**}	0.8561^{**}	$\underline{0.1856^{*}}$
TAMKOT	$\underline{0.6786}$	$\underline{0.8745^{**}}$	0.1857^{*}
GMKT	**0.6819**	**0.8960**	**0.1802**

the range of $[0, 1]$ using the assignment's maximum possible score. We then use Root Mean Squared Error (RMSE) to evaluate model prediction performance.

Comparison with Baselines: GMKT's results along with the baselines are presented in Table 3. We only run MVKM on MORF dataset due to its limitations in handling high-dimensional data with large computational time costs.

We first observe that GMKT outperforms all baseline methods, particularly in Junyi and MORF datasets, highlighting the importance of modeling both assessed and non-assessed activities for accurate student knowledge representation. The results demonstrate GMKT's effectiveness in capturing knowledge transfer between different material types and improving multi-activity student knowledge tracing through neighborhood-based and transition-aware representation learning. We also observe that the difference between GMKT and the second-best baseline is more significant in Junyi and MORF datasets. A potential explanation could be contrast in material associations and transition variability between different datasets. Contrary to GMKT which uses a complex key-value structure and neighborhood-based material representations, the second-best baseline (TAMKOT) models knowledge transfer between assessed and non-assessed materials using a simple LSTM-like structure. Hence, while the complex structure of GMKT is needed for more complex datasets, TAMKOT's performance could be adequate for the less complex ones. Particularly, in EdNet, related problems are bundled together, each problem is associated with one explanation, and students follow similar transitions between materials within bundles. So, the enhanced graph structure and complex knowledge representation may not provide much additional information in this dataset. Comparing GMKT to other two

multi-activity methods, MVKM and DMKT, it shows that GMKT significantly outperforms both of them in all datasets. This again highlights the importance of explicitly modeling knowledge transfer and activity-type transitions, as well as incorporating graph-structured information in knowledge modeling.

Moreover, the results indicate that the multi-activity variants of assessed-only methods do not consistently improve prediction performance compared to their original formulations. For instance, SAKT+M performs worse than SKAT on EdNet and Junyi datasets, while DKVMN+M performs worse than DKVMN on MORF dataset. These suggest that simply adding non-assessed activities as additional features sometimes has a negative impact on performance prediction. Nonetheless, it can improve performance when knowledge transfer between assessed and non-assessed materials is adequately modeled, like GMKT.

Ablation Studies: We conduct two sets of ablation studies to validate the impact of coarse-grained representations (the multi-activity transition graph layer) and the type objective. First, we remove the GNN component from GMKT, referred to as GMKT-G. Second, we remove the type objective term, $\lambda_o \mathcal{L}^o$, from $\mathcal{L}_{total}$, in Equation 18 (GMKT-O). According to the results in Table 4, removing either of these components has decreased performance in all datasets, indicating that neighborhood-based representations and the type objective are both necessary and can provide the most significant improvement when used together. Comparing GMKT-G and GMKT-O, we observe similar results in EdNet and Junyi. Whereas, for MORF dataset, GMKT-O outperforms GMKT-G, meaning that neighborhood-based similarities are more important than the type objective in MORF. A potential reason can be the material complexity in MORF. Each problem covers one topic in EdNet and Junyi, but each MORF assignment has multiple problems and video lectures cover multiple concepts. So, more coarse-grained representation can provide richer information about materials in MORF.

Sensitivity Analysis: To have a deeper understanding of the impact of the type objective on student performance prediction, we perform a sensitivity analysis by changing λ_o in Equation 18 while fixing all other hyperparameters to the best-learned values. The experiment results in Figure 2 show that prediction performance initially improves, but gradually decreases after reaching a certain λ_o for all datasets. This demonstrates that while adding the type objective helps in achieving higher performance, a balance is necessary between the objective function components. Additionally, while the best λ_o varies slightly for each dataset (0.1 for EdNet and Junyi and 0.05 for MORF), the overall range for optimal λ_o is small and GMKT can robustly use a similar λ_o for different datasets.

5.5 Knowledge Transfer Modeling

In this set of experiments, we focus on examining the knowledge transfer between assessed materials to non-assessed ones. Specifically, we compare the transition weight matrices $\boldsymbol{T}_{QL}$ and $\boldsymbol{T}_{LQ}$ in equation 6 to determine if the knowledge transfer from assessed to non-assessed materials differs from that of non-assessed to assessed materials. These matrices represent the weight of knowledge transfer from one memory slot to another when a student switches from one material

Table 4: Ablation study results

Methods	EdNet AUC	Junyi AUC	MORF RMSE
GMKT-G	0.6759	0.8909	0.1888
GMKT-O	0.6761	0.8911	0.1867
GMKT	**0.6819**	**0.8960**	**0.1802**

Table 5: Spearman correlation coefficients with p-values between $\boldsymbol{T}_{QL}$ and $\boldsymbol{T}_{LQ}$

	EdNet	Junyi	MORF
Correlation	0.0357	-0.0128	-0.0504
p-value	0.2531	0.4120	0.1072

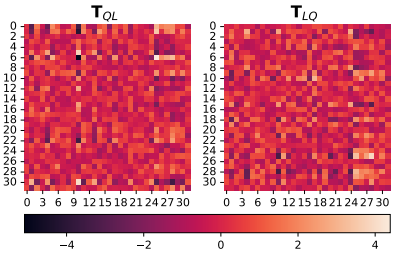

Fig. 3: Heatmaps for weight matrices $\boldsymbol{T}_{QL}$ and $\boldsymbol{T}_{LQ}$ for MORF dataset.

type to another. We flatten these matrices and calculate the Spearman correlation coefficient [31] between them. The resulting correlation coefficient and p-value are presented in Table 5, indicating that there is no significant correlation between $\boldsymbol{T}_{QL}$ (assessed to non-assessed) and $\boldsymbol{T}_{LQ}$ (non-assessed to assessed), as the correlations are small and the p-values are greater than 0.1 for all datasets. This implies that transition weights in $\boldsymbol{T}_{QL}$ and $\boldsymbol{T}_{LQ}$ are mostly different. To further investigate, we plot the heatmap of $\boldsymbol{T}_{QL}$ and $\boldsymbol{T}_{LQ}$ for the MORF dataset in Figure 3(Heatmaps for the Junyi and Ednet are in the supplementary material due to space limitations). A z-score normalization [25] is performed to $\boldsymbol{T}_{QL}$ and $\boldsymbol{T}_{LQ}$ for better visualization. As evident from the heatmap, weight matrices are considerably different from each other, indicating that knowledge transfer weights depend on the order of transition between material types. Thus, modeling knowledge transfer between different material types is sufficient.

6 Conclusions

We focused on multi-activity knowledge tracing, modeling student knowledge as they transition between various types of materials. We developed GMKT, a model with a transition-aware dynamic knowledge transfer network and a transition-aware graph neural network that captures both fine-grained and coarse-grained associations between materials. We also proposed a semi-supervised learning approach that considers both student performance and activity type objectives. Our experimental results on three real-world datasets showed that explicitly modeling transition-aware knowledge transfers, capturing coarse-grained associations by the transition-aware GNN, and adding the activity type objective, are crucial for accurately representing student knowledge and predicting their performance. Our analysis showed that student knowledge transfers between assessed and non-assessed activities depend on transition order, indicating that transition-aware models are essential for multi-activity knowledge tracing.

Acknowledgements. This paper is based upon work supported by the National Science Foundation under Grant No. 2047500.

7 Ethical Statement

This research paper aims to assess students' knowledge states through their learning activities. It can assist students or instructors in better understanding student learning, which can be used to plan students' study plans to improve students' learning efficiency, recommend useful learning materials to students, and detect knowledge gaps in students. The research team has given careful consideration to the ethical implications of collecting and processing student personal data, and the potential inference of student personal information in our work. We ensured that all data used adhered to relevant laws and regulations, and relied on the publicly available datasets that ensure to maintain the anonymity and confidentiality of any personal information that could be inferred from the data. Furthermore, we have considered the broader impact of our research on society and aimed to ensure that it has a positive impact. As a part of broader impact, we strive to make all the developed code via this research publicly available. We acknowledge that ethical considerations are crucial in scientific research, and we have taken all necessary measures to ensure that our work meets the highest ethical standards. In summary, we confirm that our work follows ethical guidelines and standards set by the scientific community.

References

1. S. Abdi. Learner models for learnersourced adaptive educational systems. 2022.
2. S. Abdi, H. Khosravi, S. Sadiq, and A. Darvishi. Open learner models for multi-activity educational systems. In *International Conference on Artificial Intelligence in Education*, pages 11–17. Springer, 2021.
3. R. Agrawal, M. Christoforaki, S. Gollapudi, A. Kannan, K. Kenthapadi, and A. Swaminathan. Mining videos from the web for electronic textbooks. In *International Conference on Formal Concept Analysis*, pages 219–234. Springer, 2014.
4. J. M. L. Andres, R. S. Baker, G. Siemens, D. Gašević, and C. A. Spann. Replicating 21 findings on student success in online learning. *Technology, Instruction, Cognition, and Learning*, 10(4):313–333, 2016.
5. H. Cen, K. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International conference on intelligent tutoring systems*, pages 164–175. Springer, 2006.
6. H. Cen, K. Koedinger, and B. Junker. Comparing two irt models for conjunctive skills. In *International Conference on Intelligent Tutoring Systems*, pages 796–798. Springer, 2008.
7. H.-S. Chang, H.-J. Hsu, and K.-T. Chen. Modeling exercise relationships in e-learning: A unified approach. In *EDM*, pages 532–535, 2015.
8. Y. Choi, Y. Lee, J. Cho, J. Baek, B. Kim, Y. Cha, D. Shin, C. Bae, and J. Heo. Towards an appropriate query, key, and value computation for knowledge tracing. In *Proceedings of the 7th ACM Conference on Learning at Scale*, pages 341–344, New York, NY, USA, 2020. ACM.
9. Y. Choi, Y. Lee, D. Shin, J. Cho, S. Park, S. Lee, J. Baek, C. Bae, B. Kim, and J. Heo. Ednet: A large-scale hierarchical dataset in education. In *International Conference on Artificial Intelligence in Education*, pages 69–73. Springer, 2020.

10. CMU DataShop. Junyi dataset. https://pslcdatashop.web.cmu.edu/Project?id=244, 2015.
11. A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278, 1994.
12. C. Cui, Y. Yao, C. Zhang, H. Ma, Y. Ma, Z. Ren, C. Zhang, and J. Ko. Dgekt: A dual graph ensemble learning method for knowledge tracing. *arXiv preprint arXiv:2211.12881*, 2022.
13. F. Drasgow and C. L. Hulin. Item response theory. *Handbook of Industrial and Organizational Psychology*, pages 577–636, 1990.
14. A. Ghosh, N. Heffernan, and A. S. Lan. Context-aware attentive knowledge tracing. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2330–2339, New York, NY, USA, 2020. ACM.
15. M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE, 2005.
16. S. Haykin. *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1994.
17. M. I. Jordan, M. J. Kearns, and S. A. Solla. *Advances in Neural Information Processing Systems 10: Proceedings of the 1997 Conference*, volume 10. Mit Press, 1998.
18. A. S. Lan, A. E. Waters, C. Studer, and R. G. Baraniuk. Sparse factor analysis for learning and content analytics. *Journal of Machine Learning Research (JMLR)*, 15(57):1959–2008, 2014.
19. Q. Liu, Z. Huang, Y. Yin, E. Chen, H. Xiong, Y. Su, and G. Hu. Ekt: Exercise-aware knowledge tracing for student performance prediction. *IEEE Transactions on Knowledge and Data Engineering*, 33(1):100–115, 2019.
20. Y. Liu, Y. Yang, X. Chen, J. Shen, H. Zhang, and Y. Yu. Improving knowledge tracing via pre-training question embeddings. *arXiv preprint arXiv:2012.05031*, 2020.
21. T. Long, Y. Liu, J. Shen, W. Zhang, and Y. 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.
22. A. S. Najar, A. Mitrovic, and B. M. McLaren. Adaptive support versus alternating worked examples and tutored problems: which leads to better learning? In *International Conference on User Modeling, Adaptation, and Personalization*, pages 171–182. Springer, 2014.
23. H. Nakagawa, Y. Iwasawa, and Y. Matsuo. Graph-based knowledge tracing: modeling student proficiency using graph neural network. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 156–163, 2019.
24. S. Pandey and G. Karypis. A self-attentive model for knowledge tracing. In *Proceedings of the 12th International Conference on Educational Data Mining*, pages 384–389. International Educational Data Mining Society, 2019.
25. S. Patro and K. K. Sahu. Normalization: A preprocessing stage. *arXiv preprint arXiv:1503.06462*, 2015.
26. P. I. Pavlik Jr, H. Cen, and K. R. Koedinger. Performance factors analysis–a new alternative to knowledge tracing. *Online Submission*, 2009.
27. C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, page 505–513, Cambridge, MA, USA, 2015. MIT Press.

28. X. Qi, R. Liao, J. Jia, S. Fidler, and R. Urtasun. 3d graph neural networks for rgbd semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 5199–5208, 2017.
29. Y. Qu, T. Bai, W. Zhang, J. Nie, and J. Tang. An end-to-end neighborhood-based interaction model for knowledge-enhanced recommendation. In *Proceedings of the 1st international workshop on deep learning practice for high-dimensional sparse data*, pages 1–9, 2019.
30. S. Sahebi, Y.-R. Lin, and P. Brusilovsky. Tensor factorization for student modeling and performance prediction in unstructured domain. *International Educational Data Mining Society*, 2016.
31. C. Spearman. The proof and measurement of association between two things. 1961.
32. N. Thai-Nghe, T. Horváth, and L. Schmidt-Thieme. Factorization models for forecasting student performance. In *Educational Data Mining 2011*. Citeseer, 2010.
33. S. Tong, Q. Liu, W. Huang, Z. Hunag, E. Chen, C. Liu, H. Ma, and S. Wang. Structure-based knowledge tracing: An influence propagation view. In *2020 IEEE international conference on data mining (ICDM)*, pages 541–550. IEEE, 2020.
34. C. Wang, S. Zhao, and S. Sahebi. Learning from non-assessed resources: Deep multi-type knowledge tracing. *International Educational Data Mining Society*, 2021.
35. X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 950–958, 2019.
36. Y. Yang, J. Shen, Y. Qu, Y. Liu, K. Wang, Y. Zhu, W. Zhang, and Y. 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.
37. L. Yao, C. Mao, and Y. Luo. Graph convolutional networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7370–7377, 2019.
38. C. K. Yeung. Deep-irt: Make deep learning based knowledge tracing explainable using item response theory. In *Proceedings of the 12th International Conference on Educational Data Mining*, pages 683–686. International Educational Data Mining Society, 2019.
39. J. Zhang, X. Shi, I. King, and D.-Y. Yeung. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th International Conference on World Wide Web*, pages 765–774, New York, NY, USA, 2017. ACM.
40. L. Zhang, X. Xiong, S. Zhao, A. Botelho, and N. T. Heffernan. Incorporating rich features into deep knowledge tracing. In *Proceedings of the 4th ACM Conference on Learning at Scale*, pages 169–172, New York, NY, USA, 2017. ACM.
41. S. Zhao, C. Wang, and S. Sahebi. Modeling knowledge acquisition from multiple learning resource types. In *Proceedings of The 13th International Conference on Educational Data Mining*, pages 313–324. International Educational Data Mining Society, 2020.
42. S. Zhao, C. Wang, and S. Sahebi. Transition-aware multi-activity knowledge tracing. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 1760–1769. IEEE, 2022.