Learning or Forgetting? A Dynamic Approach for Tracking the Knowledge Proficiency of Students

ZHENYA HUANG, QI LIU, and YUYING CHEN, University of Science and Technology of China, China
LE WU, Hefei University of Technology, China and iFLYTEK Co., Ltd, China
KELI XIAO, Stony Brook University, USA
ENHONG CHEN, University of Science and Technology of China, China
HAIPING MA, Anhui University, China
GUOPING HU, iFLYTEK Research, China

The rapid development of the technologies for online learning provides students with extensive resources for self-learning and brings new opportunities for data-driven research on educational management. An important issue of online learning is to diagnose the knowledge proficiency (i.e., the mastery level of a certain knowledge concept) of each student. Considering that it is a common case that students inevitably learn and forget knowledge from time to time, it is necessary to track the change of their knowledge proficiency during the learning process. Existing approaches either relied on static scenarios or ignored the interpretability of diagnosis results. To address these problems, in this article, we present a focused study on diagnosing the knowledge proficiency of students, where the goal is to *track* and *explain* their evolutions simultaneously. Specifically, we first devise an explanatory probabilistic matrix factorization model, *Knowledge Proficiency Tracing* (KPT), by leveraging educational priors. KPT model first associates each exercise with a knowledge vector in which each element represents a specific knowledge concept with the help of *Q*-matrix. Correspondingly, at each time, each student can be represented as a proficiency vector in the same knowledge space. Then, our KPT model jointly applies two classical educational theories (i.e., *learning curve* and *forgetting curve*) to capture the change of students' proficiency level on concepts over time. Furthermore, for improving the predictive performance, we develop an improved version of KPT, named *Exercise-correlated Knowledge Proficiency Tracing* (EKPT), by considering the connectivity among exercises with the same knowledge concepts. Finally, we apply our KPT and EKPT models to three important diagnostic tasks, including knowledge estimation, score prediction, and diagnosis result visualization. Extensive experiments on four real-world

This research was partially supported by the National Key Research and Development Program of China (Grant No. 2016YFB1000904), the Science Foundation of Ministry of Education of China & China Mobile (Grant No. MCM20170507), the National Natural Science Foundation of China (Grants No. 61922073 and No. U19A2079), and the Foundation of State Key Laboratory of Cognitive Intelligence, iFLYTEK, P.R. China (Grant No. COGOS-20190002). Zhenya Huang gratefully acknowledges the support of the China Scholarship Council (Grant No. 201806340212).

Authors' addresses: Z. Huang, Q. Liu, Y. Chen, and E. Chen (corresponding author), Anhui Province Key Laboratory of Big Data Analysis and Application, University of Science and Technology of China, Hefei, Anhui, 230027, China; emails: huangzhy@mail.ustc.edu.cn, qiliuql@ustc.edu.cn, cyy33222@mail.ustc.edu.cn, cheneh@ustc.edu.cn; L. Wu, Hefei University of Technology, Hefei, Anhui, 230009, China, State Key Laboratory of Cognitive Intelligence, iFLYTEK Co., Ltd, China; email: lewu@hfut.edu.cn; K. Xiao, Stony Brook University, NY, 11794, USA; email: keli.xiao@stonybrook.edu; H. Ma, Key Laboratory of Intelligent Computing and Signal Processing of Ministry of Education, Anhui University, Hefei, Anhui, 230039, China; email: hpma2020@163.com; G. Hu, iFLYTEK Research, Hefei, Anhui, 230088, China; email: gphu@iflytek.com.

1046-8188/2020/02-ART19 $15.00
https://doi.org/10.1145/3379507

datasets demonstrate that both of our models could track the knowledge proficiency of students effectively and interpretatively.

CCS Concepts: • **Computing methodologies** → **Factorization methods**; • **Applied computing** → **Education**; • **Information systems** → *Collaborative filtering*;

Additional Key Words and Phrases: Diagnosis, knowledge proficiency levels, educational theories

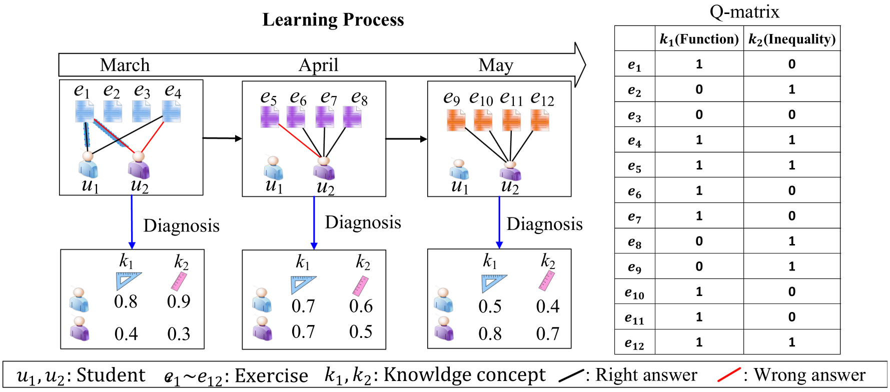
ACM Reference format:
Zhenya Huang, Qi Liu, Yuying Chen, Le Wu, Keli Xiao, Enhong Chen, Haiping Ma, and Guoping Hu. 2020. Learning or Forgetting? A Dynamic Approach for Tracking the Knowledge Proficiency of Students. *ACM Trans. Inf. Syst.* 38, 2, Article 19 (February 2020), 33 pages.
https://doi.org/10.1145/3379507

1 INTRODUCTION

Online learning systems have a long history dating back to the 1980s [55, 74] and still have witnessed the proliferation with the computer-aid technology and artificial intelligence in recent years, such as massive open online course (MOOC) platforms and online judge (OJ) systems [14, 78]. Generally, these platforms provide students with abundant learning resources (e.g., course, exercise, lecture) and enable them an open environment to learn and practice knowledge individually. Although such advantages of autonomy and convenience do attract a large number of students, researchers still have found that students are prone to lose their learning interests and show high dropout issue in practice [1, 21]. To deal with this problem, an effective solution is to provide the personalized services in online learning systems to improve students' learning experiences. Fortunately, with the accumulation of rich student-learning data nowadays, researchers have tried many data-oriented solutions on this educational issue [1, 8].

Among them, one of the key tasks is knowledge proficiency diagnosis, with the goal of discovering the latent mastery levels of students on each knowledge concept [38]. Figure 1 shows a toy example of this task. We can see that two students (i.e., u_1 and u_2) learn two knowledge concepts (i.e., k_1 (*Function*) and k_2 (*Inequality*)) by practicing on different mathematical exercises (i.e., $\{e_1, e_2, \ldots, e_{12}\}$) from March to May. Also, a Q-matrix, which is usually annotated by educational experts, denotes the relationship between exercises and concepts [15]. Specifically, the number 1 in Q-matrix means that the corresponding exercise contains the knowledge concept and 0 otherwise, e.g., exercise e_1 contains *Function*, and exercise e_4 is related to both *Function* and *Inequality*. Thus, the task of knowledge proficiency diagnosis is: Given the historical exercising record of students and the corresponding Q-matrix, we aim to analyze how much they master each concept (i.e., *Function* and *Inequality*). In fact, as these diagnosis results are beneficial to numerous applications, such as targeted knowledge training [25] and personalized exercise recommendation [47, 62], many efforts in both educational psychology and data-mining fields have been devoted to this issue: In educational psychology, cognitive diagnosis models usually characterize the knowledge proficiency of each student by a latent trait value [20] or a binary skill mastery vector [15]. Comparatively, by treating the diagnosis task as a data-mining problem of score prediction, matrix factorization techniques project students in a latent space to infer their implicit knowledge states [33]. In summary, these two types of research straightforwardly exploit the student exercising records for diagnosis. However, most of them ignore some important factors in their learning process.

In the literature, there are two main factors that have significant impacts on the learning process. On one hand, educational psychologists have long converged that the student-learning process is not static but evolves over time [70]. Inevitably, students gain and forget the knowledge they learn.



	k_1(Function)	k_2(Inequality)
e_1	1	0
e_2	0	1
e_3	0	0
e_4	1	1
e_5	1	1
e_6	1	0
e_7	1	0
e_8	0	1
e_9	0	1
e_{10}	1	0
e_{11}	1	0
e_{12}	1	1

Fig. 1. A showcase of knowledge proficiency diagnosis task for two students (u_1, u_2) on mathematical exercises ($e_1, e_2, \ldots, e_{12}$) related to two knowledge concepts (k_1 (*Function*), k_2 (*Inequality*)) from March to May. The left area contains two parts: the top part shows their learning process with different exercises, and the bottom part shows the corresponding diagnosis results of them on these two knowledge concepts over time. The right area shows the *Q*-matrix that depicts the knowledge concepts of each exercise, where each row denotes an exercise and each column stands for a knowledge concept.

Specifically, two important theories in education studies provide the fundamental ideas for modeling the knowledge proficiency of students. While the *learning curve theory* argues that students can gain the knowledge with constant trails or exercises [2], the *forgetting curve theory* suggests that students have a decreasing memory on things they have learned so that their knowledge proficiency follows a declining curve [3, 19]. Let us take Figure 1 as an example. As time goes on, student u_2 improved her proficiency levels on both two knowledge concepts with some exercises she took and learned. In contrast, student u_1's proficiency levels decreased as she did not do any exercises in April and May. Based on the above two theories, several studies from both the data-mining community [40, 63, 77] and the cognitive diagnosis area [9, 12, 48, 50, 75] have attempted to track the knowledge proficiency of students dynamically. The experimental results showed the superiority of adding the temporal information for this task. However, there are some issues that are still under exploration. In particular, data-mining models, such as the tensor factorization model [34, 63], only capture the latent factors of students over time, so these models are hard to explain the causality between specific knowledge concepts and the proficiency levels of students. In practice, the interpretability is an important factor leading to good diagnosis results [38]. Besides, in cognitive diagnosis, many knowledge tracing approaches [48, 75] consider the learning and forgetting factors as additional parameters. Related results indicate that these two factors are highly related to the exercises done by students at each time as suggested by educational experts. However, these cognitive models could not answer the question that how these educational theories would help explain the evolution of students' proficiency level on concepts over time. Therefore, our work aims to apply both learning and forgetting theories to better *track* and *explain* the knowledge proficiency of students during the learning process.

On the other hand, in the real world, students often get consistent scores on the exercises with same knowledge concepts [53, 59]. This is also a crucial factor for the diagnosis task. For instance, from Figure 1, student u_1 performed well on both exercises e_1 and e_4 in March, while student u_2 answered them incorrectly. It is easy to find that both exercises contain the same concept *Function*; hence, we could naturally conclude that u_1 mastered better on *Function* than u_2. Intuitively, this evidence is similar to the phenomenon in recommender systems [22, 27, 34, 57] and information retrieval [42, 46], where users usually show similar consumption preferences on similar items (e.g., items with similar attributes). Many studies in the literature show that the item-based collaborative algorithms have successfully improved the performance in relevant fields [26, 57, 71]. To the best of our knowledge, although some prior work have considered some skill (knowledge) relationships, such as prerequisite skill hierarchies [30], few of them has noticed the effect of directly utilizing the above connectivity among exercises covering the same concepts for diagnosing and explaining the knowledge proficiency of students. In summary, in this work, we mainly focus on addressing the following challenges for tracking student knowledge proficiency: How do we apply all of the above educational factors (i.e., learning theory, forgetting theory, and connectivity property) to the knowledge diagnosis? How do we quantitatively distinguish and interpret these factors to improve both accuracy and interpretability of the diagnosis results?

To tackle these challenges, in our preliminary work [11], we proposed an explanatory probabilistic matrix factorization model called *Knowledge Proficiency Tracing* (KPT) to track the proficiency levels of students by incorporating learning and forgetting theories. Specifically, we first associated each exercise with a knowledge vector, in which each element represented a specific concept. The *Q*-matrix, which was marked by experts to depict the relationship between exercises and knowledge concepts, was exploited as priors to generate exercise representations. Each student was also represented as a proficiency vector at each time in the same knowledge space. Then, we jointly applied both *learning curve* and *forgetting curve* theories to capture the change of each student's knowledge states over time. Therefore, KPT could well *track* and *explain* the knowledge proficiency of students during their learning process.

In this article, to improve the predictive performance of our proposed model, we further develop an improved version of KPT and propose an *Exercise-correlated Knowledge Proficiency Tracing* (EKPT) model, where we incorporate the connectivity property among exercises over knowledge concepts into our probabilistic modeling. In EKPT, we assume that students perform consistently on the exercises with same knowledge concepts. To be specific, at each time, we select a neighbor set for each exercise over its concepts, and thus EKPT could learn each knowledge vector with the influence of its neighbors. Correspondingly, the proficiency vector of each student is also updated by her performances on these neighbor exercises. Furthermore, for comprehensively verifying the effectiveness of KPT and EKPT, we introduce three practical diagnostic tasks, including knowledge estimation, score prediction and diagnosis result visualization. Finally, we conduct extensive experiments on four real-world datasets, where the experimental results demonstrate the effectiveness of both our proposed models with good accuracy and interpretability.

The rest of this article is organized as follows. In Section 2, we introduce the related work. The problem definition of tracking student knowledge proficiency is specified in Section 3. Sections 4 and 5 detail our KPT and EKPT models, respectively. In Section 6, we specify how to apply our models to three diagnosis tasks. Section 7 presents the experimental results. After that, we give an overall discussion based on this work in Section 8. Finally, conclusions are given in Section 9.

2 RELATED WORK

In the literature, our work is relevant to an important topic in educational psychology, called self-regulated learning, which has been developed over the past two decades [79]. Broadly speaking, it refers to the process by which learners personally activate and sustain cognitions, affects, and behaviors that are systematically oriented toward the success of learning goals. In terms of learning performance perspective, prior work mainly focused on the school-based or classroom-based learning environment where educators could teach students the skills necessary to lead them to become self-regulated learners by using strategies such as reciprocal teaching, open-ended tasks, and project-based learning [49]. Recently, the proliferation of online learning systems, such as MOOC, have attracted many students for self-regulated learning. However, research results showed that many students greatly struggled with it due to the absence of support and guidance from instructors, resulting in the huge problem of dropout rate [1, 21, 32]. Therefore, many scholars nowadays pay more attention on the issue of how to improve the self-regulated learning strategies in online environment [32]. Among them, one of the most important strategies is self-assessment, which suggests systems monitoring the learning process of students and telling them what skills they has and what they need. Actually, our work of tracking student knowledge proficiency has very strong connections with the self-regulated learning in terms of self-assessment strategy from technique perspective. Our main goal is to propose an interpretable model of reminding students of what knowledge they have acquired and what they have not. Our solution can help students online be motivated to conduct self-regulated learning in practice in some way.

In the following, we summarize the related model techniques into four categories, i.e., student modeling, cognitive diagnosis, dynamic learning modeling, and exercise relationship modeling.

Student modeling. The first category is student modeling [73, 77] in data-mining area, with the goal to learn the latent representations of students from their exercises. These learned representations could be applied to many tasks, such as performance prediction [71]. We can regard the obtained representations of students as their implicit knowledge proficiency. Usually, there are two types of representative techniques: factorization models [33, 40, 63] and neural networks [43, 52, 53, 59, 77]. For instance, Thai-Nghe et al. [62] leveraged matrix factorization models to map each student into a latent vector that depicted her implicit knowledge states. To capture the dynamics of student-learning process, Thai-Nghe et al. [63] proposed a tensor factorization approach by incorporating additional time dimensions over time. Recently, through establishing a bridge between knowledge concepts and neurons, researchers have developed many deep neural networks for the diagnosis task. For example, Piech et al. [52] proposed a deep knowledge tracing (DKT) model, which to the best of our knowledge, was the first attempt to utilize recurrent neural networks (e.g., RNN or LSTM) for tracing students' knowledge states. Moreover, Liu et al. [53] and Minn et al. [43], respectively, incorporated the effects of exercise content and student groups for improving the performance. Nevertheless, a common limitation of these works is that these models operate like a black box, where a certain student's knowledge states on different concepts are usually integrated into one hidden unified vector. Thus, the output of her state representation are hard to explain. That is to say, neither the latent vectors from factorization models nor the hidden layers from neural networks can correspond to any explicit knowledge concept (e.g., *Function*). In contrast, our models improve traditional probabilistic models by incorporating educational factors (i.e., learning theory, forgetting theory and connectivity property), which tells us the strengths and weaknesses of students, guaranteeing the explanatory power.

Cognitive diagnosis. Cognitive diagnosis is a crucial direction in educational psychology, which aims at discovering the proficiency of students on the defined knowledge concepts [17, 65]. Widely used approaches could be divided into two aspects: unidimensional models and multidimensional models. Among them, item response theory (IRT), as a typical unidimensional model, considered each student as a single proficiency variable (i.e., latent trait) [20]. Comparatively, multidimensional models, such as *Deterministic Inputs, Noisy-And gate* (DINA) model, characterized each student by a binary latent vector, which described whether or not she had mastered the knowledge concepts with the help of *Q*-matrix [15]. Furthermore, Liu et al. [38] proposed FuzzyCDF to quantitatively diagnose the knowledge proficiency of students by taking advantage of fuzzy system. However, to the best of our knowledge, all these methods rely on static assumptions and ignore the temporal factor for more precise diagnosis. In this work, we focus on capturing the change of students' knowledge proficiency levels during their learning process.

Dynamic learning process modeling. To explain the dynamics of students' knowledge proficiency during the learning process, educational psychologists have converged two classical theories: *Learning curve theory* argues that students can enhance their knowledge acquisition with constant trails or exercises [2] and *Forgetting curve theory* indicates that students have a decreasing memory on things they have learned as time goes on [3, 19]. Based on these two theories, researchers have attempted to develop a series of models for diagnosing the knowledge states of students from an evolving perspective. For example, some IRT-based models, such as learning factors analysis [9] and performance factors analysis [50], were proposed, which assumed that students shared the same parameters of learning rates when exercising. Furthermore, Wang et al. [70] proposed a time-series IRT model to estimate a dynamic latent trait of each student. In addition, another representative work is knowledge tracing [12, 23, 30, 31, 48, 64, 75], which is a typical sequential framework to trace students' knowledge states over time. Among them, Bayesian knowledge tracing (BKT) [12] was one of the most popular models, which assumed each student's knowledge state as a set of binary variables, where each variable represented she had "mastered" or "non-mastered" on a certain concept. It leveraged hidden Markov models (HMM) to update the concept states separately. On this basis, some extensions further considered the effects of other factors, such as individual differences [64] and prerequisite hierarchies [30]. Despite the importance of these efforts, there are still some limitations in practice: First, IRT-based models only estimate a specific variable (e.g., latent trait) for each student so that they cannot discover her proficiency levels on multiple knowledge concepts simultaneously (e.g., *Function* and *Inequality* in Figure 1). Second, knowledge tracing usually works with a learning scenario where students are allowed to keep practicing the same items for learning target concepts, which is not effective enough for a more general one (as shown in Figure 1) where students seldom repeat doing the same exercises but seek more different exercises for required concept learning. Last but not least, existing models neglect the direct evolutionary influence of both learning and forgetting factors when students are exercising, thus are hard to quantify the dynamics of their knowledge proficiency over time.

Exercise relationship modeling. Modeling exercise relationship is one of the key issues in educational psychology. On one hand, researchers often utilize *Q*-matrix to associate exercises and knowledge concepts [4, 16]. In practice, the *Q*-matrix is usually marked by experts (e.g., teacher), defining which knowledge concepts are needed for each exercise. The original inspiration for the *Q*-matrix method came from Tatsuoka et al. [61] who explored student misconceptions in basic math concepts, such as adding fractions. Recently, Sun et al. [60] and Liu et al. [37] made attempts to automatically generate the *Q*-matrix to reduce the cost of expertise. With the given *Q*-matrix, many efforts have been made to generate educational analysis, such as knowledge diagnosis [10, 16, 75], slip and guess detection [38], and learning team formation [39]. On the other hand, researchers focus on leveraging exercise relationships (e.g., prerequisite hierarchies) to predict student scores, which could help analyze the knowledge states of students. The initiative idea behind is that students usually get consistent scores on the exercises with same knowledge concepts [30, 62]. Intuitively, this kind of idea is similar to the methods used in recommender systems and information retrieval [42, 46, 71], such as item-based methods [26, 57] and model-based methods [22, 35], where users usually show the similar consumption preferences on similar items (e.g., items that they bought in the past or items with same attributes). Although some previous work explore the effects of prerequisite relationship [30] of concepts, it requires a large of experts' annotations, which is labor intensive. To the best of our knowledge, it is still under-explored to combine the connectivity of exercises for more precisely diagnosing students' proficiency levels. In this article, we focus on capturing the connectivity property among exercises over knowledge concepts into our modeling, which improves both accuracy and interpretability of the diagnosis results.

Table 1. Example: Left Table Shows a Typical Exercising Log

(a) Exercising log example

Student	Exercise	Time	Score
u_1	e_1	t_1	0
u_1	e_5	t_2	0.25
u_2	e_2	t_1	0
u_2	e_3	t_3	1
u_2	e_1	t_3	0.75
u_3	e_4	t_4	1
...	...	...	...

(b) Q-matrix example

Exercise	Knowledge concepts				
	k_1	k_2	k_3	k_4	k_5
e_1	1	0	0	0	0
e_2	0	0	1	0	0
e_3	0	0	0	1	1
e_4	0	1	0	0	0
e_5	1	0	0	0	0
...	...	...	...	...	...

Right table presents a Q-matrix.

3 PRELIMINARIES

In this section, we first formulate our problem of tracking the knowledge proficiency of students. Then, we present an overview for our solution.

3.1 Problem Statement

In general, suppose there are N students, M exercises, and K knowledge concepts in a learning system where students do exercises at different times, recorded as exercising logs (Table 1(a)). Specifically, we can represent the exercising logs as a score tensor $R \in \mathbb{R}^{N \times M \times T}$. If student i does exercise j at time t, then R_{ij}^t denotes student i's performance score on exercise j. In addition, we are also given a Q-matrix provided by educational experts, which can be represented as a binary knowledge matrix $Q \in \mathbb{R}^{M \times K}$ (Table 1(b)). If exercise j relates to knowledge concept k, then $Q_{jk} = 1$; otherwise, $Q_{jk} = 0$. In our scenario, please note that at different time, most students practice the same exercises only once, because they usually choose different exercises to learn a specific knowledge concept in general cases. For example, suppose a student tries to learn concept *Function*, she will first practice one related exercise and check whether or not she is right. If she finds the answer is wrong, then she will not practice the same exercise, since she has already known the answer. Therefore, it can be nature that she will practice another one but with the same concept for *Function* learning. For this reason, we can see that student u_1 in Table 1 learns concept k_1 by practicing different exercises e_1 and e_5 at different time.[1] Without loss of generality, the research problem can be formulated as follows:

(PROBLEM FORMULATION) *Given the score tensor R and the corresponding* Q*-matrix* Q*, our goal is twofold: (1) tracking the change of knowledge proficiency of each student and diagnosing how much she masters* K *knowledge concepts from time 1 to* T*; (2) predicting her knowledge proficiency on* K *concepts and performance scores on specific exercises at time* $T + 1$.

3.2 Solution Overview

Our solution overview is shown in Figure 2. Specifically, based on students' exercising logs and the corresponding Q-matrix, we first propose a primary *Knowledge Proficiency Tracing* (KPT) model. KPT first projects the proficiency vector of each student into a knowledge space with the help of Q-matrix prior and then combines both *Learning curve theory* and *Forgetting curve theory* for tracking her knowledge proficiency over time. Furthermore, we propose an improved *Exercise-correlated Knowledge Proficiency Tracing* (EKPT) model by incorporating the exercise connectivity

[1] There are many different scenarios in different online learning systems. We give the detailed discussion in Section 8.

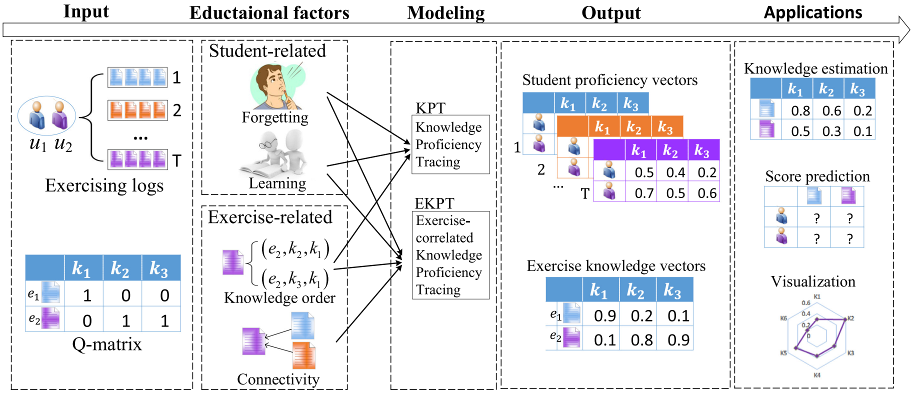


Fig. 2. The overview of our solutions.

Table 2. The Key Mathematical Notations

Notation	Description
N	the total number of students
M	the total number of exercises
T	the total number of time windows
K	the total number of knowledge concepts
R_{ij}^t	the response score of student i on exercise j at time t
U_i^t	the proficiency vector of student i on knowledge concepts at time t, $U_i^t \in \mathbb{R}^{K\times 1}$
V_j	the knowledge vector of exercise j on knowledge concepts, $V_j \in \mathbb{R}^{K\times 1}$
b_j	the difficulty bias of exercise j
α_i	the balance parameter of student i
N_{V_j}	the neighbor set of exercise j with same knowledge concepts

to improve the prediction performance. After that, we can obtain the student proficiency vectors U at different time and the exercise knowledge vectors V. At last, we apply both KPT and EKPT models to three educational tasks, i.e., estimating the knowledge proficiency (U^{T+1}) of students in the future, predicting their scores (R^{T+1}) in the future, and visualizing the diagnosis results.

In the following, we will specify the probabilistic modeling and parameter learning of KPT and EKPT, respectively. For better illustration, the key notations are summarized in Table 2.

4 KNOWLEDGE PROFICIENCY TRACING MODEL

In this section, we first introduce the *Knowledge Proficiency Tracing* (KPT) model, which contains two major steps: modeling exercise knowledge vectors V with Q-matrix prior and modeling student proficiency vectors U with both learning and forgetting theories.

4.1 Probabilistic Modeling with Priors

Generally, inspired by many existing works [26, 76], given the student exercising logs, for each student i and each exercise j, we model the conditional distributions of observed response score tensor R with student proficiency vectors and exercise knowledge vectors as

$$p(R|U,V,b) = \prod_{t=1}^{T}\prod_{i=1}^{N}\prod_{j=1}^{M}\left[\mathcal{N}\left(R_{ij}^t \middle| \left\langle U_i^t, V_j\right\rangle - b_j, \sigma_R^2\right)\right]^{I_{ij}^t}, \quad (1)$$

where $\mathcal{N}(\mu, \sigma^2)$ is a Gaussian distribution with mean μ and variance σ^2. I is an indicator tensor, where the variable I_{ij}^t equals to 1 if student i practices exercise j at time t, and vice versa. $U_i^t \in \mathbb{R}^{K \times 1}$ is the proficiency vector of student i on K knowledge concepts at time t in student matrix U^t. $V_j \in \mathbb{R}^{K \times 1}$ is the knowledge vector of exercise j in exercise matrix V, denoting the latent correlation between exercise j and K concepts. b_j is the difficulty bias of exercise j, which is widely adopted in many cognitive modeling [20]. $\langle \cdot, \cdot \rangle$ denotes the inner product of two vectors. To track the knowledge proficiency of students, we assume that their proficiency levels on K concepts change among time in Equation (1). Therefore, we specify the students' proficiency levels as a tensor presentation as: $U = \{U^1, \ldots, U^t, \ldots, U^T\}$. Given the likelihood function in Equation (1), in the following, we first explain how to embed the Q-matrix prior to model exercise matrix V, endowing each exercise knowledge vector with interpretability in which each element represents a specific knowledge concept. Then, we track student proficiency tensor U by combining *Learning curve* and *Forgetting curve* theories in the dynamic learning process.

4.1.1 Modeling V with the Q-matrix Prior. Traditional probabilistic models in Equation (1) always suffer from the low interpretation problem as the learned latent vectors are unexplainable [33, 62], i.e., each dimension in the student tensor U and the exercise vectors V cannot relate to any specific knowledge concept (e.g., *Function*), and thus we cannot remind students of their weaknesses, which is not satisfied enough [77]. For this issue, many efforts in educational psychology have been made to build an interpretable model by leveraging Q-matrix [15], i.e., letting the qth dimension in V_j embed with concept q if Q_{jq} =1. However, such traditional Q-matrix has two disadvantages in practice: (1) inevitable error or subjective bias due to manual labeling [37]; (2) the sparsity with the binary entries, which does not fit probabilistic modeling well. To mitigate these existing issues, we refine Q-matrix by utilizing a partial-order-based method [54]. In this way, we can reduce both disadvantages and associate each exercise with all K knowledge concepts. More formally, for exercise j, we first define its partial order $>_j^+$ of all concepts as

$$\begin{aligned} &\forall p, q \in K, p \neq q, \text{if } Q_{jq} = 1 \text{ and } Q_{jp} = 0 \Rightarrow q >_j^+ p, \\ &\forall p, q \in K, p \neq q, \text{if } Q_{jq} = 1 \text{ and } Q_{jp} = 1 \Rightarrow q \not>_j^+ p, \\ &\forall p, q \in K, p \neq q, \text{if } Q_{jq} = 0 \text{ and } Q_{jp} = 0 \Rightarrow q \not>_j^+ p. \end{aligned} \tag{2}$$

Here, if a knowledge concept q is marked as 1 in Q-matrix, i.e., $Q_{jq} = 1$, then we assume that this concept q is more relevant to exercise j than all other concepts with mark 0. Please note that we cannot infer the comparability of knowledge concepts with the same mark. Using this partial order $>_j^+$, we can transform the original Q-matrix into a comparable triplet set $D_Q \subseteq (M \times K \times K)$ as

$$D_Q = \left\{ (j, q, p) \middle| q >_j^+ p \right\}. \tag{3}$$

Therefore, D_Q is not as sparse as Q-matrix and can more accurately capture the pairwise relationships between two knowledge concepts (q, p) for exercise j. Then our goal is to learn the latent exercise matrix $V \in \mathbb{R}^{M \times K}$ in Equation (1) by incorporating this order set D_Q. Along this line, the Bayesian process of finding the correct partial order on all pairs of knowledge concepts (q, p) for all exercise vectors V turns to maximizing the following posterior probability over D_Q as

$$p(V | D_Q) \propto p(D_Q | V) \times p(V). \tag{4}$$

Here, all exercises are presumed to be marked independently by experts in Q-matrix. We also assume the ordering of each pair of knowledge concepts (q, p) for a specific exercise j is independent of the ordering of every other pair [54]. Hence, the above exercise-specific likelihood function

$p(D_Q|V)$ in Equation (4) can be defined as follows:

$$p(D_Q|V) = \prod_{(j,q,p)\in D_Q} p\Big(q >_j^+ p\Big|V_j\Big). \tag{5}$$

To guarantee the order relation on exercise vector V_j, we define the individual probability that knowledge concept q is more relevant to exercise j than concept p with logistic sigmoid as

$$p\Big(q >_j^+ p\Big|V_j\Big) = \frac{1}{1 + \mathrm{e}^{-(V_{jq}-V_{jp})}}. \tag{6}$$

Besides, following the traditional Bayesian treatment, we assume that the general prior density $p(V)$ in Equation (4) follows a zero-mean Gaussian prior:

$$p\Big(V\Big|\sigma_V^2\Big) = \prod_{j=1}^{M} \mathcal{N}\Big(V_j\Big|0, \sigma_V^2\mathbf{I}\Big). \tag{7}$$

In summary, we can formulate the log posterior distribution Equation (4) of exercise matrix V over order set D_Q by combining Equations (5), (6), and (7) as

$$\begin{aligned}\ln p(V|D_Q) &= \ln p(D_Q|V) + \ln p(V) = \ln \prod_{(j,q,p)\in D_Q} p\Big(>_j^+ \Big|V\Big) + \ln\prod_{j=1}^{M}\mathcal{N}\Big(V_j\Big|0,\sigma_V^2\mathbf{I}\Big)\\ &= \sum_{j=1}^{M}\sum_{q=1}^{K}\sum_{p=1}^{K} I\left(q >_j^+ p\right) \ln \frac{1}{1+\mathrm{e}^{-\left(V_{jq}-V_{jp}\right)}} - \sum_{j=1}^{M}\frac{1}{2\sigma_V^2}||V||_F^2,\end{aligned} \tag{8}$$

where $I(q >_j^+ p)$ is a indicator that equals to 1 if the triplet (j,q,p) exists in order set D_Q.

4.1.2 Modeling U with Dynamic Educational Theories. Now, we specify how to model the evolution of student latent tensor U. As mentioned before, during the dynamic learning process of each student, there are two widely accepted theories in educational psychology that could guide us in the modeling process: (1) *Learning curve* [2] depicts the knowledge learned by the students can be enhanced with several exercising trails; (2) *Forgetting curve* [3] hypothesizes that students will remember less and less about what they have learned so that their proficiency levels on knowledge concepts will gradually decline over time.

Combining these two theories as priors, we assume that a certain student's current knowledge proficiency is mainly influenced by two underlying reasons: (1) The more exercises she does, the higher level of proficiency on the related knowledge she will get; (2) The longer the time passes, the more knowledge she will forget. Formally, we model the two effects of each student's knowledge proficiency at time $t = 2, 3, \ldots, T$ as

$$\begin{aligned}&p\Big(U_i^t\Big) = \mathcal{N}\Big(U_i^t\Big|\bar{U_i^t}, \sigma_U^2\mathbf{I}\Big), \quad \text{where } \bar{U_i^t} = \Big\{\bar{U_{i1}^t}, \bar{U_{i2}^t}, \ldots, \bar{U_{iK}^t}\Big\},\\ &\bar{U_{ik}^t} = \alpha_i L_{ik}^t(*) + (1-\alpha_i)F_{ik}^t(*), \quad \textit{s.t.}\ 0 \le \alpha_i \le 1,\end{aligned} \tag{9}$$

where $U_i^t \in \mathbb{R}^{K\times 1}$, the proficiency vector of student i at time t, follows a Gaussian distribution with mean $\bar{U_i^t}$ and variance $\sigma_U^2\mathbf{I}$. It consists of student i's proficiency on all K knowledge concepts, where U_{ik}^t describes her proficiency level on concept k at time t. $L_{ik}^t(*)$ is the learning factor, denoting how much student i will learn knowledge k at time t after several exercising trails and $F_{ik}^t(*)$ is the forgetting factor, indicating her remaining knowledge level on concept k at time t. α_i is a non-negative parameter that balances these two factors to capture student i's learning characteristics. Intuitively, if student i has a large α_i, she may be diligent, and thus $L_{ik}^t(*)$, instead of $F_{ik}^t(*)$, affects her future knowledge proficiency more significantly. Otherwise, the forgetting factor $F_{ik}^t(*)$ plays more important role.

Now our goal turns to how to define the specific learning and forgetting factors, i.e., $L_{ik}^t(*)$ and $F_{ik}^t(*)$, respectively. Specifically, the learning factor $L_{ik}^t(*)$ captures the proficiency growth of student i on concept k with a number of exercising trails. Taking this intuition into consideration, in the literature [2], there are various specific learning curve forms, such as log-linear curve, exponential curve and hyperbolic curve. In this article, we select the 2-parameter hyperbolic learning curve for implementation as the hyperbolic one proves to be more robust for better fitness results [45].[2] Formally, we model the learning factors $L_{ik}^t(*)$ as follows:

$$L_{ik}^t(*) = U_{ik}^{t-1} \frac{Df_{ik}^t}{f_{ik}^t + r}, \tag{10}$$

where f_{ik}^t denotes the frequency (the number) of exercises related to knowledge concept k practiced by student i at time t. r and D are two hyper-parameters, which control the magnitude and multiplier of the growth, respectively.

Comparatively, the forgetting factor $F_{ik}^t(*)$ depicts the decline of student i's proficiency level on knowledge concept k as time goes by [3]. There are also various optional curve forms, such as power curve and exponential curve. Here, we select the typical exponential form for specifying the forgetting factor $F_{ik}^t(*)$ as

$$F_{ik}^t(*) = U_{ik}^{t-1} \mathrm{e}^{-\frac{\Delta t}{S}}, \tag{11}$$

where Δt is the time interval between time window $t-1$ and time window t, and S is a hyper-parameter that denotes the strength of memory.

Moreover, at the initial time $t = 1$, since we do not know the knowledge level of each student, we assume a zero-mean Gaussian distribution of students' knowledge proficiency at that time. Then, we summarize the prior on user latent tensor U with both educational theories as

$$p\left(U \middle| \sigma_U^2, \sigma_{U1}^2\right) = \prod_{i=1}^{N} \mathcal{N}\left(U_i^1 \middle| 0, \sigma_{U1}^2 \mathbf{I}\right) \prod_{t=2}^{T} \mathcal{N}\left(U_i^t \middle| \bar{U}_i^t, \sigma_U^2 \mathbf{I}\right). \tag{12}$$

4.2 Model Learning

We summarize the graphical representation of the proposed latent model in Figure 3, where the shaded and unshaded variables indicate the observed and latent variables, respectively. With this graphical model defined above, our goal is to learn the parameters $\Phi = [U, V, \alpha, b]$, where $\alpha = [\alpha_i]_{i=1}^N$ and $b = [b_j]_{j=1}^M$. Specifically, we can formulate the maximum posterior distribution of Equation (1) over parameters Φ by combining Equations (1), (4), and (9) as follows:

$$p(U, V, \alpha, b | R, D_Q) \propto p(R|U, V, b) \times p(U|\alpha) \times p(V|D_Q). \tag{13}$$

Maximizing the log posterior of the Equation (13) is equivalent to minimizing the following objective by incorporating the inferences with Equations (8) and (12):

$$\begin{aligned}
\min_{\Phi} \mathcal{E}(\Phi) = {} & \frac{1}{2} \sum_{t=1}^{T} \sum_{i=1}^{N} \sum_{j=1}^{M} I_{ij}^t \left(\hat{R}_{ij}^t - R_{ij}^t\right)^2 \\
& - \lambda_P \sum_{j=1}^{M} \sum_{q=1}^{K} \sum_{p=1}^{K} I\left(q >_j^+ p\right) \ln \frac{1}{1 + \mathrm{e}^{-(V_{jq} - V_{jp})}} + \frac{\lambda_V}{2} \sum_{j=1}^{M} ||V_j||_F^2 \\
& + \frac{\lambda_U}{2} \sum_{t=2}^{T} \sum_{i=1}^{N} ||\overline{U_i^t} - U_i^t||_F^2 + \frac{\lambda_{U1}}{2} \sum_{i=1}^{N} ||U_i^1||_F^2,
\end{aligned} \tag{14}$$

[2] Please note that comparing the performance of different learning curves and forgetting curves is not the main focus in this work, readers can refer to References [2, 3] for more details.

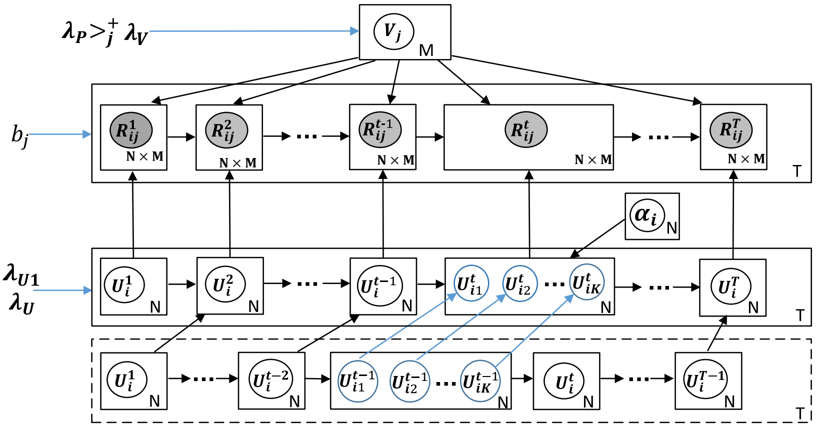


Fig. 3. Graphical representation of KPT, where the shaded and unshaded symbols indicate the observed and latent variables, respectively.

where $\lambda_P = \sigma_R^2$, $\lambda_U = \frac{\sigma_R^2}{\sigma_U^2}$, $\lambda_{U1} = \frac{\sigma_R^2}{\sigma_{U1}^2}$, and $\lambda_V = \frac{\sigma_R^2}{\sigma_V^2}$. Among them, λ_P is a tradeoff coefficient between the score prediction loss and the partial order loss, and λ_U is a coefficient that measures how student's knowledge proficiency changes over time. λ_{U1} and λ_V are regularization parameters for student proficiency matrix at time 1 and the knowledge matrix of exercises, respectively.

Although the coupling of parameters Φ makes the above loss function of Equation (14) not-convex, we can achieve a local minimum of it by performing gradient descent on each parameter iteratively [71]. Specifically, the derivatives of each parameter are

$$\begin{aligned}\nabla_{U_{ik}^t} = &\sum_{j=1}^{M} I_{ij}^t\left(\hat{R}_{ij}^t - R_{ij}^t\right)V_{jk} + \mathcal{I}[t=1]\lambda_{U1}U_{ik}^1 + \mathcal{I}[t\geq 2]\lambda_U\left(\overline{U_{ik}^t} - U_{ik}^t\right)\\ &+ \lambda_U\left(\overline{U_{ik}^{(t+1)}} - U_{ik}^{(t+1)}\right)\left((1-\alpha_i)\mathrm{e}^{-\frac{\Delta t}{S}} + \alpha_i\frac{Df_{ik}^t}{f_{ik}^t + r}\right),\end{aligned} \tag{15}$$

$$\begin{aligned}\nabla_{V_{jk}} = &\sum\nolimits_{t=1}^{T}\sum\nolimits_{i=1}^{N} I_{ij}^t\left(\hat{R}_{ij}^t - R_{ij}^t\right)U_{ik}^t + \lambda_V V_{jk} - \lambda_P\sum\nolimits_{p=1}^{K} I\left(k >_j^+ p\right)\frac{\mathrm{e}^{-\left(V_{jk}-V_{jp}\right)}}{1+\mathrm{e}^{-\left(V_{jk}-V_{jp}\right)}}\\ &- \lambda_P\sum\nolimits_{q=1}^{K} I\left(q >_j^+ k\right)\frac{-\mathrm{e}^{-\left(V_{jq}-V_{jk}\right)}}{1+\mathrm{e}^{-\left(V_{jq}-V_{jk}\right)}},\end{aligned} \tag{16}$$

$$\nabla_{\alpha_i} = \lambda_U\sum_{t=2}^{T}\sum_{k=1}^{K}\left(\overline{U_{ik}^t} - U_{ik}^t\right)\left(U_{ik}^t\left(\frac{Df_{ik}^t}{f_{ik}^t + r} - \mathrm{e}^{-\frac{\Delta t}{S}}\right)\right), \tag{17}$$

$$\nabla_{b_j} = \sum_{i=1}^{M} I_{ij}^t\left(\hat{R}_{ij}^t - R_{ij}^t\right), \tag{18}$$

where $\mathcal{I}[x]$ is an indicator function that equals to 1 if x is true and 0 otherwise.

For the updating step, as there are no constraints on U, V, and b, we can update them directly by using Stochastic Gradient Descent (SGD) method [7]. With the bound constraints of α_i, a local minimum can be found by the Projected Gradient (PG) method [36]. Specifically, for each $\alpha_i \in [0,1]$ the PG method updates the current solution α_i^k in kth iteration to α_i^{k+1} by the following rule:

$$\alpha_i^{k+1} = P\left[\alpha_i^k - \eta\nabla_{\alpha_i}\right], \quad P(\alpha_i) = \begin{cases}\alpha_i & \text{if} \quad 0 \leq \alpha_i \leq 1,\\ 0 & \text{if} \quad \alpha_i < 0,\\ 1 & \text{if} \quad \alpha_i > 1,\end{cases} \tag{19}$$

where η is so-called learning rate in the updating process. In summary, we give the training algorithm of KPT model in Algorithm 1.

ALGORITHM 1: Parameter Learning of the KPT Model

```
Initialize $U, V, \alpha$ and $b$;
while not converged do
    for $i = 1, 2, \ldots N$ do
        for $t = 1, 2, \ldots, T$ do
            for $k = 1, 2, \ldots, K$ do
                Fix $V, \alpha, b$, update $U_{ik}^t$ by Equation (15) using SGD;
        Fix $U, V, b$, update $\alpha_i$ by Equation (17) and Equation (19) using PG;
    for $j = 1, 2, \ldots, M$ do
        for $k = 1, 2, \ldots, K$ do
            Fix $U, \alpha, b$, update $V_{jk}$ by Equation (16) using SGD;
        Fix $U, V, \alpha$, update $b$ by Equation (18) using SGD;
Return $U, V, \alpha$ and $b$;
```

Time Complexity. KPT costs most time in computing the knowledge proficiency of each student and balancing parameters. Suppose there are r non-empty entries in response score tensor R, then the average number of score records of each student in each time window are $t_r = \frac{r}{N \times T}$. In each iteration, the time complexity is $O(N \times T \times K \times t_r = O(K \times r))$ for student proficiency vectors U, $O(K \times r)$ for exercise knowledge vectors V, and $O(r)$ for the balance parameters. Thus, the total complexity of parameter learning in each iteration is $O(K \times r)$.

5 EXERCISE-CORRELATED KNOWLEDGE PROFICIENCY TRACING MODEL

In the KPT model, we have already incorporated the educational learning theories (i.e., *learning curve* and *forgetting curve*) for tracking the knowledge proficiency of each student. Students are presented as explicit proficiency vectors in which each element reflects how much they have learned about the relevant knowledge concept (e.g., *Function*), guaranteeing the interprebility of the diagnosis results. However, in the real world, students may practice very few exercises compared with the huge exercise space [21, 59]. So, KPT could hardly track the knowledge proficiency and predict the performance score of a certain student if she just practices few exercises at each time. Therefore, to alleviate this problem and improve the predictive performance of our KPT model, we further consider the connectivity relationship among exercises. Please recall Figure 1, where each exercise is related to some underlying knowledge concepts, which means the exercises with the same knowledge concepts are similar in the knowledge space. Therefore, students may get consistent scores on these knowledge-based exercises [59], where this evidence is beneficial for the diagnosis. For instance, from Figure 1, we could conclude that u_1 masters better than u_2 in May, because u_1 answers both exercises e_1 and e_4 right but u_2 fails. Motivated by the above analysis, in this section, we extend the current KPT model and propose a novel *Exercise-correlated Knowledge Proficiency* (EKPT) model by incorporating this connectivity property into our probabilistic modeling.

5.1 Modeling V with Exercise Connectivity

To be specific, in the KPT model, we view the exercise prior $p(V)$ in Equation (4) as a traditional zero-mean Gaussian distribution, i.e., Equation (7). This treatment is classfical and widely used in many probablistic Bayesian modeling process, which takes the advantage of avoiding overfitting [44]. Nevertheless, it only models the feature representation of each exercise individually, but ignores the relationship among exercises. Thus, in this EKPT model, we carefully handle the connectivity relationship into the modeling with the help of Q-matrix.

Mathematically, for each exercise j, we first define a neighbor set N_{V_j}, which contains its similar exercises with the same knowledge concepts: $N_{V_j} = \{l|k \in j \cap l, l \in V, k \in K\}$. Then, we assume that the knowledge vector of exercise j is directly influenced by this neighbor set N_{V_j} as follows:

$$V_j = \sum_{l \in N_{V_j}} w(j,l) \times V_l + \theta_V, \theta_V \sim \mathcal{N}\left(0, \sigma_V^2\right). \tag{20}$$

In Equation (20), each exercise j's knowledge vector is composed of two terms. The first term characterizes the group feature of the neighbor exercises, where $w(j,l)$ describes the weight influence of each neighbor l on j. The second term emphasizes the uniqueness of each exercise knowledge vector, which could diverge from N_{V_j} to an extent. The divergence is controlled by the variance parameter σ_V^2. In this article, we straightforwardly specify the weight $w(j,l)$ as the equal influence, which is the average value of neighbor set N_{V_j}. Therefore, Equation (20) can be transformed as

$$V_j = \frac{1}{|N_{V_j}|} \sum_{l \in N_{V_j}} V_l + \theta_V, \theta_V \sim \mathcal{N}\left(0, \sigma_V^2\right). \tag{21}$$

Using this simple mathematical transformations with Equation (21), the exercise prior $p(V)$ in Equation (7) of KPT turns to the following equation:

$$p(V) = \prod_{j=1}^{M} \mathcal{N}\left(\frac{1}{|N_{V_j}|} \sum_{l \in N_{V_j}} V_l, \sigma_V^2\right). \tag{22}$$

Combining Equations (4), (5), (6), and (22), we could incorporate connectivity relationship to model exercise vectors V by transforming the log posterior distribution in Equation (8) as

$$\begin{aligned} \ln p(V|D_Q) &= \ln \prod_{(j,q,p) \in D_Q} p\left(>_j^+ \middle| V\right) p(V) \\ &= \sum_{j=1}^{M} \sum_{q=1}^{K} \sum_{p=1}^{K} I\left(q >_j^+ p\right) \ln \frac{1}{1 + \mathrm{e}^{-\left(V_{jq} - V_{jp}\right)}} - \sum_{j=1}^{M} \frac{1}{2\sigma_V^2} ||V - \frac{1}{|N_{V_j}|} \sum_{l \in N_{V_j}} V_l||_F^2. \end{aligned} \tag{23}$$

As a result, we could project each exercise into the knowledge space by incorporating the connectivity relationship. In the EKPT model, we also embed two educational learning theories (i.e., *learning curve* and *forgetting curve*) for tracking student proficiency in the same knowledge space, which is the same as KPT model, i.e., Equation (9). Through this modeling, we could find that at each time, each student's proficiency vector is not only influenced by her performance scores on the exercises she has done but also reflected by the information from the similar exercises in the knowledge space. It takes two types of advantages. First, EKPT model follows the natural evidence that students perform consistently on the exercises with same knowledge concepts, which helps predict students' performance more accurately. Second, by incorporating the exercise connectivity into the modeling, EKPT model could address the sparsity problem when students just leave very few learning trajectories, and therefore, improving the diagnosis results.

5.2 Model Learning

We also summarize the graphical representation of the EKPT model in Figure 4, where the shaded and unshaded symbols indicate the observed and latent variables, respectively. Comparing it with

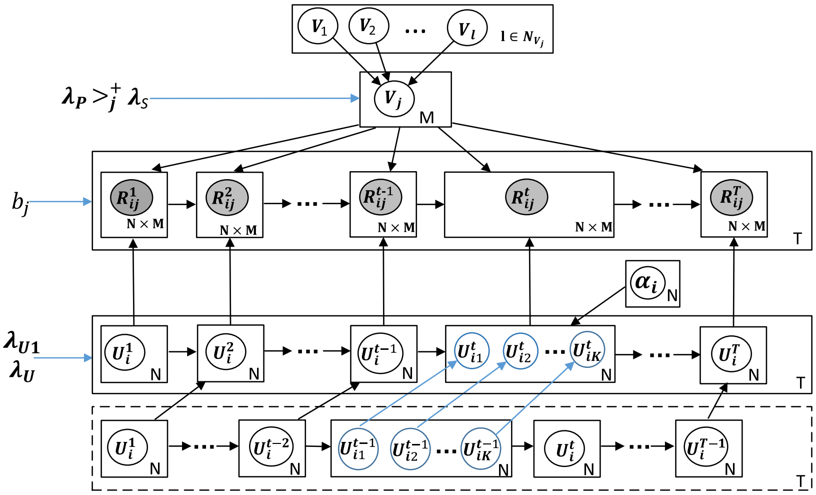


Fig. 4. Graphical representation of EKPT, where the shaded and unshaded symbols indicate the observed and latent variables, respectively.

Figure 3, we can easily find that the only difference between EKPT and KPT is that V_j in EKPT is restricted by both the partial order parameters (λ_p, $>_j^+$) and the parameters of its knowledge-based exercise set (λ_S, $N_{v_j} = \{V_1, V_2, \ldots, V_l\}$). Therefore, to learn the parameters $\Phi = [U, V, \alpha, b]$ in Equation (13), the objective function of EKPT could be updated as follows:

$$\begin{aligned}\min_{\Phi} \mathcal{E}(\Phi) = {}& \frac{1}{2}\sum_{t=1}^{T}\sum_{i=1}^{N}\sum_{j=1}^{M} I_{ij}^t\left(\hat{R}_{ij}^t - R_{ij}^t\right)^2 \\ & - \lambda_P \sum_{j=1}^{M}\sum_{q=1}^{K}\sum_{p=1}^{K} I\left(q >_j^+ p\right) \ln \frac{1}{1 + e^{-(V_{jq} - V_{jp})}} + \frac{\lambda_S}{2}\sum_{j=1}^{M} ||V_j - \frac{1}{|N_{V_j}|}\sum_{l \in N_{V_j}} V_l||_F^2 \\ & + \frac{\lambda_U}{2}\sum_{t=2}^{T}\sum_{i=1}^{N} ||\overline{U_i^t} - U_i^t||_F^2 + \frac{\lambda_{U1}}{2}\sum_{i=1}^{N} ||U_i^1||_F^2, \end{aligned} \tag{24}$$

where $\lambda_P = \sigma_R^2$, $\lambda_U = \frac{\sigma_R^2}{\sigma_U^2}$, $\lambda_{U1} = \frac{\sigma_R^2}{\sigma_{U1}^2}$, and $\lambda_S = \frac{\sigma_R^2}{\sigma_V^2}$. Please note that the above objective function of EKPT adds an additional restricted regular term on exercise vectors V, which is weighted by parameter λ_S. Correspondingly, the derivatives of parameters $\{U, \alpha, b\}$ remain unchanged with Equations (15), (17), and (18), respectively. However, the updated gradient of exercise vector parameter V in Equation (16) turns to the formulation as follows:

$$\begin{aligned}\nabla_{V_{jk}} = {}& \sum\nolimits_{t=1}^{T}\sum\nolimits_{i=1}^{N} I_{ij}^t\left(\hat{R}_{ij}^t - R_{ij}^t\right)U_{ik}^t + \lambda_S\left(V_{jk} - \frac{1}{|N_{V_j}|}\sum_{l \in N_{V_j}} V_{lk}\right) \\ & - \lambda_P \sum\nolimits_{p=1}^{K} I\left(k >_j^+ p\right)\frac{e^{-(V_{jk} - V_{jp})}}{1 + e^{-(V_{jk} - V_{jp})}} - \lambda_P \sum\nolimits_{q=1}^{K} I\left(q >_j^+ k\right)\frac{-e^{-(V_{jq} - V_{jk})}}{1 + e^{-(V_{jq} - V_{jk})}}, \end{aligned} \tag{25}$$

where $\mathcal{I}[x]$ is also an indicator function that equals to 1 if x is true. In summary, we summarize the training algorithm of EKPT in Algorithm 2.

ALGORITHM 2: Parameter Learning of the EKPT Model

```
Initialize $U$, $V$, $\alpha$ and $b$;
while not converged do
    for $i = 1, 2, \ldots N$ do
        for $t = 1, 2, \ldots, T$ do
            for $k = 1, 2, \ldots, K$ do
                Fix $V, \alpha, b$, update $U_{ik}^t$ by Equation (15) using SGD;
        Fix $U, V, b$, update $\alpha_i$ by Equation (17) and Equation (19) using PG ;
    for $j = 1, 2, \ldots, M$ do
        for $k = 1, 2, \ldots, K$ do
            Fix $U, \alpha, b$, update $V_{jk}$ by Equation (25) using SGD;
        Fix $U, V, \alpha$, update $b$ by Equation (18) using SGD;
Return $U$, $V$, $\alpha$ and $b$;
```

Time Complexity. Compared with the EKT model, EKPT costs most of time in computing the knowledge proficiency of each student and the exercise correlations. Suppose there are r non-empty entries in the response score tensor R and s knowledge-based exercise pairs ($r \ll N \times M, s \ll M \times M$), then the average number of score records of each student in each time window are $t_r = \frac{r}{N \times T}$, and the average number of similar connections of each exercise are $t_s = \frac{s}{M}$. In each iteration, the time complexity is $O(N \times T \times K \times t_r) = O(K \times r)$ for student proficiency vectors U, $O(K \times r + M \times t_s) = O(K \times r + s)$ for exercise knowledge vectors V, and $O(r)$ for the balance parameters. Thus, the total complexity of parameter learning in each iteration is $O(K \times r + s)$.

6 APPLICATIONS

Diagnosing the knowledge proficiency of each student is important for many educational applications. In this section, we will introduce how to apply our models (KPT and EKPT) to three practical diagnostic tasks, i.e., knowledge estimation, score prediction, and diagnosis result visualization.

6.1 Tasks Description

Traditional education reports usually tell students about their scores or rankings in the corresponding scenarios, e.g., classroom or online systems. However, these results are not satisfied enough, because we cannot notice what the students have learned well and what they have not. As many literatures [38, 55, 56] suggested, a deep and interpretable diagnosis result would have many benefits in online learning systems. Therefore, with our proposed KPT and EKPT models, we introduce three diagnostic tasks for practical applications:

1. **Knowledge Proficiency Estimation** is the task of estimating the probability that a student masters a certain knowledge concept at next time [38, 77]. It could benefit both student users and system creators. On one hand, students are not satisfied if we just recommend the exercises to them, since they usually wonder why they should do the exercises. Therefore, it is significant to imply the future knowledge proficiency of students as this could directly remind them of strengths and weaknesses. After that, they may be motivated to devote more energy to the unfamiliar knowledge, which can improve their learning efficiency. On the other hand, system creators can design more proactive services for students based on the estimation results. For example, it is a good choice to recommend some relevant

exercises based on their weaknesses instead of letting them self-search resources [56], where these results could help convince students.

2. **Student Score Prediction** addresses the probability that a certain student will correctly answer an exercise in the future [52, 63]. Usually, each student will apply the relevant knowledge to answer the exercises, and the score she get depends on her knowledge states. In practice, low prediction error implies that the model has accurately discovered the exercises that are easier/harder for students. With the prediction results, we could provide personalized exercise recommendation for students, avoiding them practicing too hard/easy exercises, which can save their time.
3. **Diagnosis Results Visualization** means specifying the strengths and weaknesses of each student on each knowledge concept in an interpretative and attractive way [6, 47]. Many platforms (e.g., Knewton.com) hold that it is necessary to quantitatively track the knowledge levels of students during their learning process in time, where the visualization of diagnosis results could help them trust the systems of understanding the change of their knowledge states and improve many problems such as self-regulation and self-reflection [5, 68]. Moreover, this application also benefits many students with different background, such as students in primary and secondary school [18], junior high school [69], web-based platform [24], leading many services like personalized learning, remedy planning and course design, and so on.

6.2 Knowledge Proficiency Estimation

As mentioned before, students are usually curious about how much they could master each knowledge concept in the future. Thus, in this subsection, we specify how to estimate the knowledge proficiency of each student at next time.

Specifically, after training our KPT or EKPT model, we could obtain proficiency vectors of a certain student i at each historical time, i.e., $U_i = \{U_i^1, U_i^2, \ldots, U_i^T\}$, and her individual learning parameter, i.e., α_i. With her current states on each knowledge concept U_i^T, we combine both her learning factor ($L_{ik}^t(*)$) and forgetting factor ($F_{ik}^t(*)$) to estimate her knowledge proficiency at time $T+1$ based on Equations (10) and (11) as follows:

$$\begin{aligned} \hat{U}_i^{(T+1)} &= \left\{\hat{U}_{i1}^{(T+1)}, \hat{U}_{i2}^{(T+1)}, \ldots, \hat{U}_{iK}^{(T+1)}\right\}, \\ \hat{U}_{ik}^{(T+1)} &\approx \alpha_i U_{ik}^T \frac{Df_{ik}^{T+1}}{f_{ik}^{T+1}+r} + (1-\alpha_i) U_{ik}^T \mathrm{e}^{-\frac{\Delta(T+1)}{S}}, \end{aligned} \tag{26}$$

where $\hat{U}_i^{(T+1)}$ denotes the estimated proficiency level of student i at time $T+1$ on all K knowledge concepts, where $\hat{U}_{ik}^{(T+1)}$ denotes her probability of mastering concept k at time $T+1$. $\Delta(T+1)$ describes the time interval between $T+1$ and her last exercising time about knowledge concept k. $f_{ik}^{(T+1)}$ is the frequency number of exercises related to concept k practiced by her at time $T+1$.

6.3 Student Score Prediction

It is valuable to predict how well each student will perform on exercises in the future. Usually, the effectiveness of tracking students' knowledge proficiency in the past can be validated by predicting student scores at time $T+1$. Here, we give our solution for prediction as follows.

With the trained KPT or EKPT model, we can first obtain the knowledge vector V_j of each exercise j in the knowledge space and the corresponding difficulty parameter b_j. Then, we estimate the proficiency vector U_i^{T+1} of each student i at time $T+1$, following Equation (26). At last, we predict student scores on exercises at time $T+1$, i.e., the predicted probability $\hat{R}_{ij}^{(T+1)}$ of student

i correctly answering exercise j at time $T+1$, following the inner product $\langle\cdot,\cdot\rangle$ of each student's proficiency vector and each exercise's knowledge vector as

$$\hat{R}_{ij}^{(T+1)} \approx \left\langle U_i^{(T+1)}, V_j \right\rangle - b_j. \tag{27}$$

Note that the output of Equation (27) is not between 0 and 1, and we can simply revise them as

$$\hat{R}_{ij}^{(T+1)} = \begin{cases} \hat{R}_{ij}^{(T+1)} & \text{if} \quad 0 \le \hat{R}_{ij}^{(T+1)} \le 1, \\ 0 & \text{if} \quad \hat{R}_{ij}^{(T+1)} < 0, \\ 1 & \text{if} \quad \hat{R}_{ij}^{(T+1)} > 1. \end{cases} \tag{28}$$

6.4 Diagnosis Results Visualization

As we state earlier, a good model for tracking students' knowledge proficiency should guarantee two aspects of effectiveness, which is necessary for convincing students. First, it should ensure high accuracy, which means that it can well capture the change of a certain student's knowledge states so that it can precisely predict her performance and estimate her knowledge proficiency in the future. Second, it is of great significance to provide interpretable diagnosis results, which could attract students, since they can get timely feedbacks during the learning process.

Unlike traditional data-mining models (e.g., matrix factorization) with latent representations of students and exercises, which usually cannot describe definite meanings, our KPT and EKPT models are guided by many learning factors (i.e., learning curve, forgetting curve, connectivity relationship), obtaining the meaningful results at different times. In the experimental section, we will visualize the change of students' knowledge proficiency by our models with user study. Moreover, we will discuss the connectivity relationship of exercises in the knowledge space by EKPT.

7 EXPERIMENTS

In this section, we conduct extensive experiments to evaluate the performance of our proposed KPT and EKPT models. Specifically, we first describe the datasets and introduce the experimental setup (Sections 7.1 and 7.2). Then, we demonstrate the effectiveness of our models on three educational tasks (Section 7.3). At last, we provide detailed analyses about our models (Section 7.4).

7.1 Experimental Datasets

In the experiments, we used four real-world datasets, namely, Math1, Math2, Assist, and Adaptive, respectively. Among them, Math1, Math2 were collected from daily assessment records of high school students on mathematics problems. Adaptive was collected from an online learning system called Zhixue, which provides students self-exercising on many subjects. Here, we chose the mathematical log for the experiments, because its volume was the largest in the system.[3] In addition, Assist (short for Assistments) was a public dataset 2009–2010 "Non-skill builder,"[4] which also record the mathematics logs of students in an online tutor [48].

In Math1, Math2 and Adaptive, each dataset contains the exercising logs of students (Table 1(a)) with the timestamps and a given Q-matrix (an example is shown in Table 1(b)) labeled by high school teachers. For data preprocessing, we first treated each month as a time window, and thus there were 4, 10, 7 time points in datasets Math1, Math2, Adaptive, respectively. Then, for data splitting, we used the data till time T for model training, i.e., $T = 3, T = 9, T = 6$ in Math1, Math2, Adaptive, respectively, and the records at the last time point were remained for testing.

[3] Datasets Math1, Math2 and Adaptive are provided by iFLYTEK Co., Ltd. Zhixue's website: http://www.zhixue.com.
[4] https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010-data.

Table 3. The Statistics of the Four Datasets

Dataset	Math1	Math2	Assist	Adaptive
Training logs	521,248	347,424	263,327	229,848
Testing logs	74,464	18,312	43,888	38,308
# of students	9,308	1,306	7197	3,217
# of exercises	64	280	3211	411
# of time windows	4	10	7	7
# of knowledge concepts	12	13	20	12
Avg. knowledge concepts per exercise	1.15	1.3215	1.5073	1.06

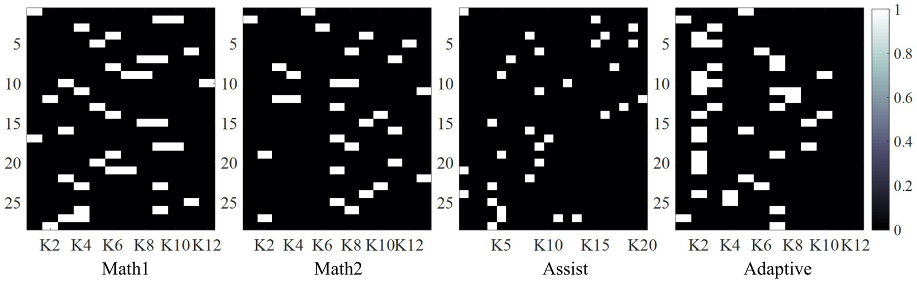


Fig. 5. Q-matrix of four datasets.

As for Assist, we preprocessed the original dataset as follows: (1) We selected the 20 most frequent knowledge concepts from Assist for the experiments, because in our scenario of knowledge proficiency tracking, the knowledge concepts that learned by students require a high coverage at different time. In other words, if a student just learned one concept once in the past, we could hardly capture the change of her learning on it. (2) The tutor system of Assist allowed students to repeat doing the same exercises. In the experiment, we just took their first-attempt responses on each exercise as records, because students might get some “hints” if they made many trials,[5] which may be unfair for our task. (3) Assist only recorded the order (no exact time) of student’s exercising history. For example, for a certain student u_1, we just knew she first practiced exercise e_1 and then answered e_2, but did not know which month she answered e_1 and e_2. For this scenario, we sorted her exercising trajectories with the given order and divided the order sequence into seven parts. Then, we marked each part of with a time window, i.e., her first log part is marked as time 1, the second one is time 2, and so on. Therefore, we got seven time windows in Assist. Then, we adopted the first six order logs for model training, and the last one was used for testing.

In summary, Table 3 lists the basic statistics of four datasets. To better understand each dataset, Figure 5 further gives the preview of four Q-matrices (we only show the subset of several exercises for better illustration), where each row of each subfigure denotes an exercise and each column stands for a certain knowledge concept (the white entry means the exercise is related to the concept). An interpretation example about the Q-matrix in Math1 dataset is shown in Table 4, which contains 5 exercises and their related knowledge concepts. From both Figure 5 and Table 4, we can see that most exercises relates to less than two knowledge concepts, indicating that Q-matrices are very sparse.

[5] Some issues are related to this scenario like “gaming factor” detecting, details will be discussed in Section 8.

Table 4. A Practical Example of Q-matrix in Math1

Exercise	knowledge concepts
e_1	Function
e_2	Function, Set
e_3	Function, Derivative, Inequality
e_4	Solid geometry, Trigonometric function
e_5	Propositional logic

7.2 Experimental Setup

7.2.1 Settings of KPT and EKPT. There are some hyper-parameters in our KPT and EKPT models, and they should be initialized in the experiments. First, we introduce the parameter settings of $L^t_{ik}(*)$ (Equation (10)) and $F^t_{ik}(*)$ (Equation (11)), i.e., the *learning curve* and *forgetting curve*, respectively. Specifically, for learning factor $L^t_{ik}(*)$, we set $D = 2$ to control the multiplier of growth and the average frequency among all knowledge concepts r as 4, 9, 6, 12 in Math1, Math2, Assist, Adaptive, respectively. For forgetting factor $F^t_{ik}(*)$, we set memory strength S as 5 in all datasets to fit the forgetting curve. Second, we specify several regularization parameters. Please note that a difference between KPT (Equation (14)) and EKPT (Equation (24)) is the regularization for exercise prior, i.e., λ_V in KPT and λ_S in EKPT. Therefore, in both models, we set $\lambda_{U1} = 0.01$, λ_U is set to be 3, 1, 2, 1.5 in Math1, Math2, Assist, Adaptive, respectively, and λ_P is set to be 1.5, 1, 2, 2 in Math1, Math2, Assist, Adaptive, respectively. Furthermore, for KPT model, we set $\lambda_V = 0.01$ in all datasets, and for EKPT model, λ_S is set to be 0.02, 0.01, 0.005, 0.1 in Math1, Math2, Assist, Adaptive, respectively. Specifically, we will discuss the sensitivity of parameters in Section 7.4.2.

7.2.2 Baseline Approaches. To compare the performance of our proposed KPT and EKPT models, we borrow some baselines from various perspectives. The details of them are as follows:

- *IRT*: IRT is a cognitive diagnosis method of modeling students' latent trait and exercises' parameters by a logistic-like function [17].
- *DINA*: DINA is a cognitive diagnosis method of modeling each student's knowledge proficiency by a binary vector with Q-matrix [15].
- *PMF*: PMF is a probabilistic matrix factorization method that projects students and exercises into low-rank latent factors [63].
- *BKT*: BKT is a kind of Hidden Markov Model (HMM), which models students' latent knowledge state as a set of binary variables and determines when a knowledge concept has been learned [31].
- *LFA*: LFA is an improved IRT model that assumes students share the same parameters of learning rate during their learning process [9].
- *DKT*: DKT is a recent state-of-the-art deep-learning-based model that incorporates recurrent neural network to model each student's knowledge states with an unified hidden vector during the learning process [52]. Here, we implemented DKT with both RNN and GRU architectures, which denoted as DKT-RNN, and DKT-GRU, respectively. The input at one step of a certain student is one-hot encodings of all exercises she practices to adapt our scenario.
- *QMIRT*: QMIRT is a variant of basic IRT model, where we extend the latent trait value of each student in IRT to a multidimensional knowledge proficiency vector with our proposed partial order prior of Q-matrix (Equation (8)).
- *QPMF*: QPMF is a variant of basic PMF model. We incorporate our proposed partial order prior of Q-matrix (Equation (8)) into PMF to endow the latent dimensions of projected

Table 5. Characteristics of the Baselines and Our Models

Model	Data Source				Application			Dynamic Explanation?
	Q-matrix	Multi-Skill	Repeating	Time	Knowledge Estimation	Score Prediction	Visualization	
IRT [17]	×	×	×	×	×	√	×	×
DINA [15]	√	√	×	×	√	√	√	×
PMF [63]	×	×	×	×	×	√	×	×
BKT [31]	√	×	√	√	√	√	√	√
LFA [9]	√	√	√	√	×	√	×	√
DKT [52]	×	√	√	√	×	√	×	√
QMIRT	√	√	×	×	√	√	√	×
QPMF	√	√	×	×	√	√	√	×
KPT	√	√	×	√	√	√	√	√
EKPT	√	√	×	√	√	√	√	√

vectors with explicit knowledge concepts. Particularly, QPMF is also a simplified model of KPT that does not consider the factors of learning and forgetting.

Specifically, the chosen baselines are all widely used in the educational psychology area (IRT, DINA, BKT, LFA) and data-mining community (PMF, DKT). The two variants (QMIRT, QPMF) are adopted to highlight the effectiveness of our proposed improved *Q*-matrix based on partial order method. One step further, all these baselines can be categorized into static diagnostic models (IRT, DINA, PMF, QMIRT, QPMF) and the dynamic ones (LFA, BKT, DKT). For better illustration, we summarize the characteristics of these models in Table 5.

In the following experiments, we utilized the open source to implement the BKT model[6] and all other models were implemented by ourselves using Python. We conducted all experiments on a Linux server with four 2.0 GHz Intel Xeon E5-2620 CPUs and 100 G memory. For fairness, all parameters in these baselines are tuned to have the best performances.

7.3 Experimental Results on Three Tasks

In this subsection, we compare the performance on three educational tasks introduced in Section 6 to demonstrate the effectiveness and interpretation of our proposed KPT and EKPT models.

7.3.1 Knowledge Proficiency Estimation. The first important ability of our models is to estimate the knowledge proficiency of each student in the future. To evaluate the effectiveness of our models, i.e., whether or not the estimation of knowledge proficiency at next time $T+1$ are good, we conduct several experiments in the following.

In fact, it is not easy to directly evaluate the knowledge state, since there is no general way to get the actual value of student's proficiency levels on knowledge concepts. As an alternative, following Reference [28], we adopt a ranking-based evaluation experiment. Intuitively, if student a masters better than student b on a specific knowledge concept k at time $T+1$ (calculated by Equation (26)), then she will have a higher probability to get correct answers to the exercises related to concept k than student b at that time. We adopt *Degree of Agreement* (DOA) [28] metric to evaluate this ranking performance. Particularly, for a specific knowledge k, the DOA result on

[6] https://github.com/IEDMS/standard-bkt.

k is defined as

$$DOA(k) = \sum_{j=1}^{M} I_{jk} \sum_{a=1}^{N} \sum_{b=1}^{N} \frac{\delta\left(U_{ak}^{T+1}, U_{bk}^{T+1}\right) \cap \delta\left(R_{aj}^{T+1}, R_{bj}^{T+1}\right)}{\delta\left(U_{ak}^{T+1}, U_{bk}^{T+1}\right)}, \tag{29}$$

where U_{ak}^{T+1} is proficiency level of student a on knowledge concept k at time $T+1$. R_{aj}^{T+1} is student a's response score on exercise j at time $T+1$. $\delta(x, y)$ is an indicator function, where $\delta(x, y) = 1$ if $x > y$. I_{jk} is another indicator function, where $I_{jk} = 1$ if exercise j contains knowledge concept k. Generally, DOA value ranges from 0 to 1. The larger the DOA is, the better performance the results have. Furthermore, we also average DOA(k) of all knowledge concepts for measuring the overall effectiveness on this task, which is denoted as DOA-Avg:

$$DOA - Avg = \frac{1}{K} \sum_{k=1}^{K} DOA(k). \tag{30}$$

For model comparisons, we choose DINA, QMIRT, QPMF, and BKT as baselines, because all other models (i.e., IRT, LFA, PMF, DKT) mentioned before are unexplainable for the diagnosis. To be specific, we cannot relate each dimension of student latent vectors in PMF, the single value of student latent trait in IRT and LFA, and the unified student knowledge state vector in DKT with any explicit knowledge concept (e.g., Function).

Table 6 illustrates the experimental results of all models on four datasets for estimating knowledge proficiency of students ("Avg" is the DOA-Avg results of all knowledge concepts and others show the DOA results of each knowledge concept). Please note that we do not give the results of DINA models on Assist and Adaptive datasets, because DINA has a high restriction that it cannot handle the sparse data in practice [15]. From the results, both EKPT and KPT clearly perform better on all datasets, followed by QPMF and QMIRT, which indicates that our models could well estimate knowledge proficiency by incorporating the factors of learning process of students (i.e., learning curve, forgetting curve, and connectivity relationship). Among them, EKPT performs the best generally. It still outperforms others in most cases even if the Assist dataset is extremely sparse (nearly 99%). Besides, we also observe that traditional cognitive diagnosis model DINA does not perform well, indicating that the static model is unsuitable for tracking students' knowledge states over time. Last but not least, we can see that BKT, as a dynamic model, does not perform as well as EKPT and KPT. This observation demonstrates the effectiveness of incorporating both educational theories including *learning curve* and *forgetting curve*.

7.3.2 Student Score Prediction. The second educational task is to predict student scores in the future, where we evaluate the predictive performance of our models. With trained KPT or EKPT models, we use Equations (27) and (28) to predict whether or not a student get the correct answer to a specific exercise at time $T+1$. In this experiment, we select all the baselines mentioned in Section 7.2.2 for comparison. Besides, we use the widely used *mean absolute error* (MAE) and *root mean square error* (RMSE) as the evaluation metrics [44].

Figure 6 shows the overall results of all models for predicting student scores. For the same reason, we do not give the results of DINA model on Assist and Adaptive datasets. Specifically, there are several observations from the figure: First, EKPT performs the best on all four datasets, followed by KPT, which indicates it is effective to take the connectivity relationship of exercises into modeling for the prediction. The results of both EKPT and KPT also show the rationality of considering both learning and forgetting factors. Second, the variants QMIRT and QPMF outperform traditional IRT and PMF, which demonstrates the effectiveness of incorporating the partial order method based on Q-matrix. Third, EKPT, KPT, and LFA, as dynamic models, perform better than

Table 6. Knowledge Proficiency Estimation Performance on Each Knowledge Concept

(a) Math1

K	Models					
	EKPT	KPT	QPMF	QMIRT	DINA	BKT
K1	**0.807**	0.798	0.565	0.595	0.524	0.558
K2	**0.751**	0.733	0.576	0.621	0.473	0.623
K3	**0.830**	0.827	0.614	0.629	0.497	0.523
K4	**0.769**	0.752	0.581	0.675	0.486	0.565
K5	**0.799**	0.791	0.559	0.723	0.476	0.578
K6	**0.844**	0.838	0.730	0.766	0.485	0.628
K7	**0.851**	0.842	0.697	0.634	0.520	0.697
K8	**0.799**	0.784	0.699	0.657	0.498	0.617
K9	**0.796**	0.771	0.609	0.712	0.501	0.645
K10	0.813	**0.834**	0.597	0.515	0.489	0.503
K11	**0.796**	0.786	0.608	0.631	0.478	0.617
K12	0.811	**0.842**	0.532	0.641	0.523	0.645
Avg	**0.806**	0.799	0.614	0.650	0.496	0.601

(b) Math2

K	Models					
	EKPT	KPT	QPMF	QMIRT	DINA	BKT
K1	**0.806**	0.804	0.743	0.754	0.517	0.568
K2	**0.761**	0.757	0.632	0.659	0.534	0.753
K3	0.816	**0.818**	0.761	0.723	0.510	0.669
K4	**0.701**	0.688	0.733	0.734	0.534	0.711
K5	**0.901**	0.891	0.703	0.668	0.474	0.553
K6	**0.704**	0.699	0.547	0.653	0.489	0.644
K7	**0.791**	0.791	0.677	0.722	0.483	0.730
K8	**0.761**	0.726	0.722	0.659	0.523	0.668
K9	**0.754**	0.736	0.558	0.541	0.507	0.567
K10	**0.674**	0.652	0.639	0.650	0.511	0.614
K11	0.869	**0.888**	0.836	0.692	0.522	0.630
K12	**0.807**	0.798	0.737	0.794	0.498	0.528
K13	**0.825**	0.813	0.797	0.804	0.453	0.633
Avg	**0.782**	0.774	0.699	0.696	0.504	0.636

(c) Assist

K	Models				
	EKPT	KPT	QPMF	QMIRT	BKT
K1	**0.781**	0.766	0.683	0.685	0.557
K2	**0.718**	0.7161	0.640	0.626	0.503
K3	**0.749**	0.705	0.662	0.657	0.546
K4	**0.671**	0.667	0.648	0.548	0.525
K5	**0.699**	0.689	0.639	0.621	0.584
K6	**0.696**	0.695	0.594	0.529	0.511
K7	**0.703**	0.684	0.583	0.655	0.601
K8	**0.675**	0.670	0.597	0.589	0.592
K9	0.651	**0.659**	0.600	0.593	0.557
K10	**0.667**	0.664	0.650	0.581	0.577
K11	0.620	**0.656**	0.626	0.594	0.548
K12	**0.648**	0.643	0.619	0.568	0.546
K13	0.653	**0.656**	0.628	0.647	0.628
K14	0.653	0.635	**0.685**	0.648	0.681
K15	**0.638**	0.627	0.627	0.634	0.638
K16	**0.651**	0.640	0.555	0.625	0.573
K17	**0.632**	0.627	0.601	0.625	0.583
K18	**0.667**	0.652	0.526	0.627	0.561
K19	0.623	**0.625**	0.616	0.611	0.554
K20	0.596	0.602	0.609	0.613	**0.642**
Avg	**0.670**	0.664	0.619	0.614	0.575

(d) Adaptive

K	Models				
	EKPT	KPT	QPMF	QMIRT	BKT
K1	**0.742**	0.732	0.656	0.645	0.578
K2	**0.799**	0.780	0.756	0.740	0.609
K3	**0.796**	0.793	0.752	0.736	0.592
K4	**0.804**	0.802	0.737	0.638	0.679
K5	**0.812**	0.808	0.597	0.632	0.552
K6	**0.818**	0.812	0.659	0.648	0.547
K7	**0.821**	0.815	0.587	0.668	0.687
K8	**0.824**	0.818	0.624	0.591	0.532
K9	**0.824**	0.809	0.704	0.692	0.645
K10	**0.823**	0.819	0.730	0.776	0.732
K11	**0.830**	0.820	0.658	0.685	0.702
K12	**0.809**	0.792	0.709	0.693	0.690
Avg	**0.809**	0.801	0.681	0.679	0.629

those with static assumption (IRT, DINA, PMF), which demonstrates that it is more effective to track students' knowledge proficiency from an evolving perspective. Forth, BKT does not perform well on this task. This is probably because BKT focuses on the scenario that students keep doing the same exercises. However, most students in our data just practice a specific exercise only once, and thus the length of students' exercise sequences are not long enough for training BKT. An interesting finding is that dynamic DKT-RNN(GRU), although utilizing the state-of-the-art deep neural networks for modeling, still performs unsatisfied enough. We guess two possible reasons for this observation. First, our data volume may not support DKTs, because deep models usually

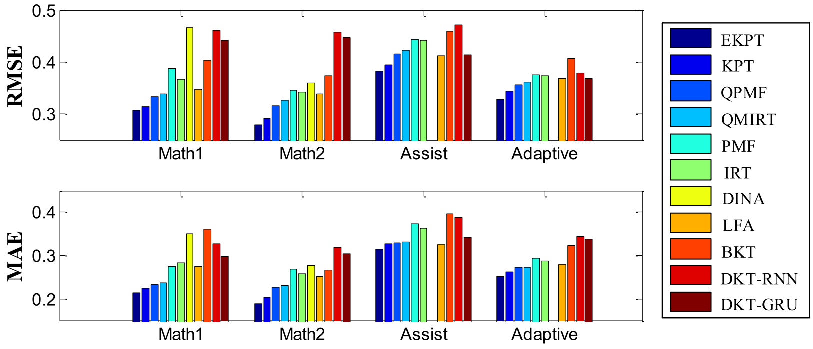


Fig. 6. Student score prediction performance.

Fig. 6. Student score prediction performance.

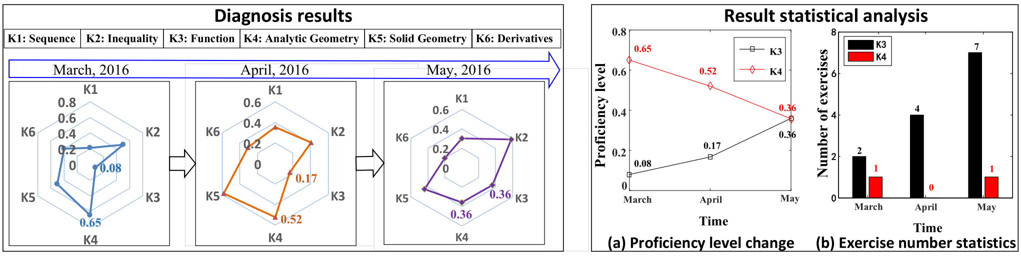


Fig. 7. User study: diagnosis results of a case student from March to May, 2016, in Math2. Left box shows her knowledge proficiency levels on six knowledge concepts (the name of concepts are listed above) in the 3 months. Right box analyzes the diagnosis result on her two concepts: The line chart illustrates the change of her proficiency levels on knowledge $K3$ and $K4$. The bar charts counts the number of exercises related to both concepts she does in each month; i.e., she practices four exercises related to $K3$ in April.

have too many parameters to be optimized. Second, the exercising sequences of students (consistent with the number time windows in Table 3) in our data are not longer enough for training DKTs, where the RNN-based models cannot capture the learning dynamics. In summary, all these evidences demonstrate the effectiveness and rationality of the proposed factors in our modeling (i.e., learning curve, forgetting curve, and connectivity relationship).

7.3.3 Diagnosis Result Visualization. As mentioned in Section 6, our proposed KPT and EKPT models have a good ability to track the knowledge proficiency of a certain student in an interpretable way, based on her proficiency vectors U at different times.

Figure 7 provides a user study of visualizing the diagnosis results of a case student on six knowledge concepts in three months in Math2 (we only show six knowledge concepts for better illustration). From the figure, she continuously makes progress on *Function* ($K3$) from March (0.08) to May (0.36), 2016 with possible learning factor, because she practices the increasing number (i.e., 2, 4, 7) of exercises related to *Function* in three months. In contrast, her proficiency levels on *Analytic geometry* ($K4$) declines (from 0.65 to 0.36) over time. We notice that she only tries very few relevant exercises (i.e., 1, 0, 1) at each time, and thus she may forget the *Analytic geometry* knowledge. These observations imply that she needs a timely review on *Analytic geometry*. Based on the visualization, the system could provide more personalized training services for her in practice.

Table 7. The Parameter Updating Time of All Models (Min)

Dataset	Time	Stastic Models			Dynamic Models				Variants		Our Models	
		IRT	DINA	PMF	BKT	LFA	DKT (RNN)	DKT (GRU)	QMIRT	QPMF	KPT	EKPT
Math1	Each	0.022	0.316	0.023	/	0.024	0.403	0.479	0.036	0.025	0.083	0.101
	Total	1.960	18.05	1.833	1.516	2.483	22.867	195.375	3.647	2.535	8.334	11.66
Math2	Each	0.011	0.616	0.021	/	0.012	0.122	0.157	0.016	0.012	0.067	0.073
	Total	1.051	57.28	1.283	0.581	1.152	7.720	10.435	1.603	1.589	7.334	7.738
Assist	Each	0.015	/	0.033	/	0.026	1.594	3.207	0.283	0.265	0.467	0.735
	Total	2.320	/	4.951	1.275	2.991	73.324	147.522	26.38	29.94	47.13	77.15
Adaptive	Each	0.013	/	0.029	/	0.015	0.273	0.338	0.105	0.110	0.233	0.453
	Total	2.154	/	3.466	1.017	1.942	11.734	12.522	8.412	10.45	24.73	48.92

7.4 Model Analysis

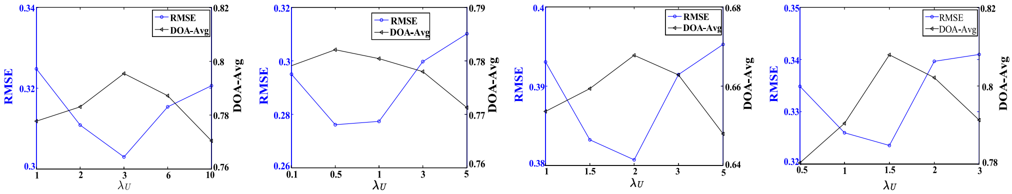
In this subsection, we further deeply analyze the important properties of our proposed models. Specifically, we discuss them from the following three aspects, i.e., the computational performance, the sensitivity of crucial parameters and the exercise connectivity analysis.

7.4.1 Computational Performance. We conduct the following experiments to understand the efficiency of our proposed models compared with all baselines. For fair comparison, we run all of them on the same platform. Since most models need to iteratively calculate the parameters, we list the parameter updating time of each iteration and total iterations in Table 7.[7] As we can see from this table, IRT model costs the least time as it only needs to optimize unidimensional parameters of each student and exercise [17]. PMF ranks the second, because it is the basic model from data mining for prediction. Compared to IRT and PMF, the variants QMIRT and QPMF spend more time by adding the partial order of Q-matrix. As for the dynamic models, BKT runs faster than LFA, since it only tracks each knowledge concept separately [31]. On average, both our KPT and EKPT need more time for training, because they all incorporate both learning and forgetting factors in the modeling. Furthermore, since EKPT associates the exercise relationship, it has to select the neighbor set (N_{V_j} Equation (21)) of each exercise in each iteration for training, costing the time. Next, we find that DINA, though a static diagnosis model, costs much time, because its complexity is exponential with the number of knowledge concepts [15]. At last, DKT-RNN(GRU) are the most time-consuming in most cases as the deep neural networks usually need more time to optimize parameters. In summary, although our proposed models cost some more time for training, yet as mentioned in Section 5, the complexities of them in each iteration are still linear with the number of student exercising records. Thus, in practical applications, we could train KPT and EKPT offline and store the parameters in the server. Then, we could get real-time results in the online stage. In summary, we argue that both KPT and EKPT have the most satisfied results for educational tasks.

7.4.2 Sensitivity of Parameters. We now discuss the parameter sensitivities in our proposed models. For better illustration, in the following, we only show the results of the EKPT model. (The detailed discussion about the KPT model could be found in our preliminary work [11]).

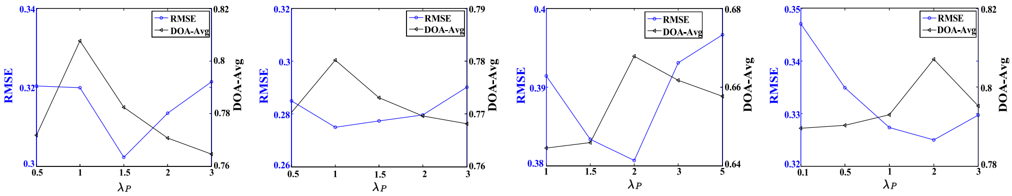
Specifically, in the EKPT model, there are four parameters playing crucial roles: λ_{U1}, λ_U, λ_P, and λ_S. Among them, λ_{U1} is the regularization parameter of students' vectors of knowledge proficiency at time $T = 1$. Since λ_{U1} has a similar form to that in PMF model, we tune it on PMF and set the

[7] Please note, we cannot record the updating time of each iteration in BKT with the public implementation.



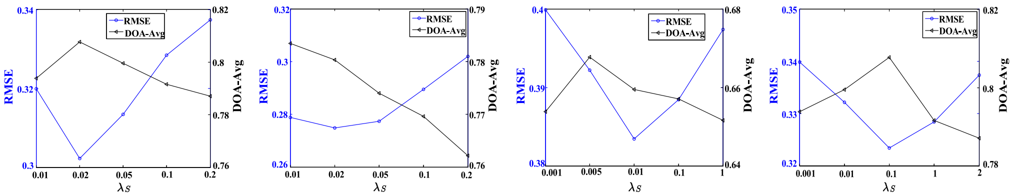
(a) Math1 (b) Math2 (c) Assist (d) Adaptive

Fig. 8. The impact of λ_U on four datasets.



(a) Math1 (b) Math2 (c) Assist (d) Adaptive

Fig. 9. The impact of λ_P on four datasets.



(a) Math1 (b) Math2 (c) Assist (d) Adaptive

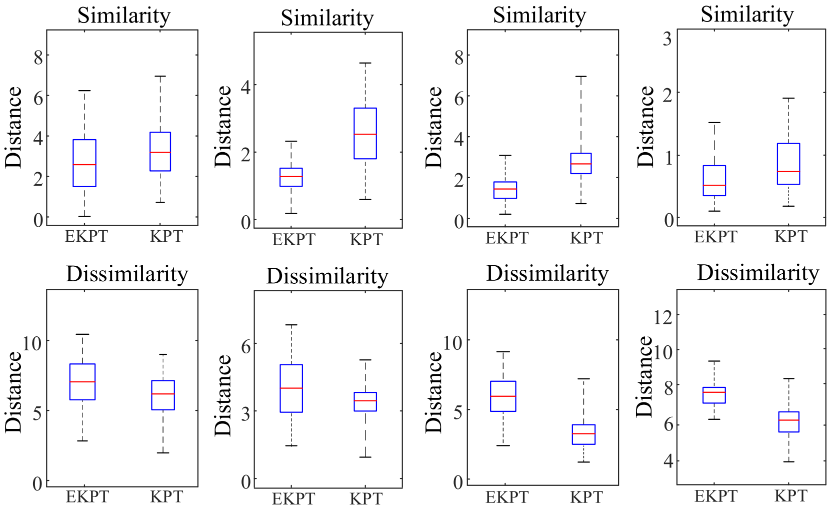
Fig. 10. The impact of λ_S on four datasets.

value under the best performance of PMF. In the following, we report the setting of parameters λ_U, λ_P and λ_S with the evaluation metrics of RMSE and DOA-Avg on both knowledge estimation task and score prediction task.

λ_U regularizes that students learn and forget knowledges from time to time. Figure 8 visualizes the performance with the increasing values of λ_U from 1, 0.1, 1, 0.5 to 10, 5, 5, 3 in datasets Math1, Math2, Assist, Adaptive, respectively. As we can see from these figures, as λ_U increases, the performances of EKPT firstly increase but decrease when λ_U surpasses 3, 1, 2, 1.5 in the corresponding datasets. Therefore, we set λ_U = 3, 1, 2, 1.5 in Math1, Math2, Assist, Adaptive for obtaining the best results.

λ_P controls how much the EKPT model is restricted by the partial order based on Q-matrix. Figure 9 shows the performance with the varying of parameter λ_P. We observe that the values of λ_P impacts both two educational tasks. Specifically, as λ_P increases, the performance of EKPT increases at first and reaches the perk when λ_P = 1.5, 1, 2, 2 in Math1, Math2, Assist, Adaptive, respectively. Given these observations, we set λ_P = 1.5, 1, 2, 2 in the corresponding datasets.

Another important parameter of EKPT model is λ_S, which restricts the impacts of exercise connectivity. As shown in Figure 10, it has the similar property to λ_U and λ_P. As a result, we set λ_S = 0.02, 0.05, 0.01, 0.1 in Math1, Math2, Assist, Adaptive, respectively, because the performance of EKPT achieves the best when it reaches the corresponding value.



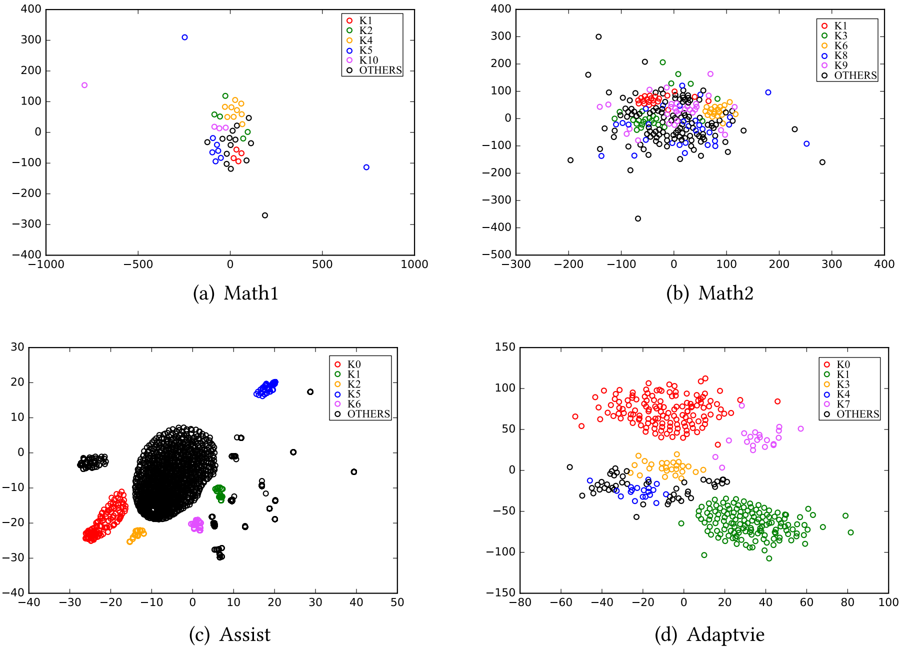
(a) Math1 (b) Math2 (c) Assist (d) Adaptive

Fig. 11. Results comparison of exercise relationship with EKPT and KPT in all datasets.

7.4.3 Exercise Relationship Analysis. As mentioned before, by incorporating exercise relationship into the modeling, EKPT could further discover the underlying connectivity between exercises, which leads to the best performance of our proposed models. Here, we compare EKPT and KPT models with the output of their exercise knowledge vectors, i.e., the parameters V, to show this property of exercise relationship on all four datasets. In each model, we first select the group exercises with same knowledge concept, e.g., *Function*. Then, for these group exercises, we combine them in pairs and calculate their Euclidean distance using their knowledge vectors V, which denoted as "Similarity." Comparatively, we also select the exercises with different knowledge concepts and calculate their pairwise Euclidean distance as "Dissimilarity." The overall results of this analysis are shown in Figure 11. We can see that exercises with the same knowledge concepts in EKPT have smaller distances than those exercises in KPT in the knowledge space. Comparatively, exercises with different knowledge concepts in EKPT are farther apart than those in KPT. Based on this evidence, we can conclude that EKPT has a good ability to combine the connectivity relationship among exercises over the knowledge concepts.

Moreover, we visualize the learned knowledge vectors of exercises, i.e., V, using a more straightforward observation to demonstrate their relationship in EKPT model. Specifically, we highlight the most frequent 5 knowledge concepts and their corresponding exercises in all datasets (we only highlight parts of knowledge concepts and group others into "OTHERS" category for better illustration). Then, to help our visualization, we adopt $t - SNE$ [41][8] program, which is commonly used for the visualization of high-dimensional data, to reduce the dimensionality of each exercise vector (each row in V) to a 2D data. Finally, we label each exercise with its knowledge concepts using different colors. The clustering results on all datasets are shown in Figure 12. In all datasets, we find that exercises with the same knowledge concepts are easier to be grouped, since they are more close in the knowledge space. Therefore, our EKPT model could naturally follow the connectivity relationship of exercises during the modeling process.

[8] https://lvdmaaten.github.io/tsne/.



(a) Math1

(b) Math2

(c) Assist

(d) Adaptvie

Fig. 12. Clustering results of exercises in all datasets. Exercises with the top five frequent knowledge concepts are distinguished by different colors. Other exercises are marked in one "OTHERS" category.

8 DISCUSSION

In this section, we comprehensively discuss the advantages and some possible research directions in the future. In this article, we describe the problem of tracking students' knowledge proficiency in an educational domain-specific way. Here, we take advantage of both learning curve and forgetting curve theories from educational psychology for modeling student-learning process and explore connectivity relationship for associating different exercises. Moreover, the detailed diagnosis result visualization of user study demonstrates that our models can provide an interpretable way to explain the change of students' knowledge proficiency levels by considering their exercising frequencies as time goes by. Although all the data we used are related to mathematics, it is worth mentioning that our models can be easily generalized to other subjects, especially science-related subjects like physics, as long as we can collect the exercising data of students with the knowledge settings. For example, we can track students' knowledge levels of physics if we can collect the log data of them and the related knowledge concepts of physics.

Meanwhile, there are still some important issues that can be explored in the future. First, as we mention earlier, although nowadays online learning systems provide students with an open environment for sharing the resources to help personalized learning without geographical restrictions, they still suffer from the important problems like huge dropout rate and low engagement [1, 21, 32, 79]. In practice, how to explore more self-regulated learning strategies, such as goal setting, strategic planning [32], to alleviate this issue is still a very open direction. For example, we can design more reward mechanisms while monitoring the behaviors of students to automatically change the service strategies for supporting their learning [29]. Moreover, we can also design more visualizations for supporting students with different background like primary schools, high schools, and so on [5, 18].

Second, there are various learning scenarios for students online. For example, some (second) language-learning systems, such as Duolingo,[9] can push students practice one item again and again (e.g., training pronunciation) [58]. Online judge systems like Codeforces[10] allow students to resubmit answers (i.e., multiple-attempt response) until they pass the assessment [78]. In these scenarios, our proposed methods may be not effective, since both methods just focus on the final response results without considering specific behaviors of students. Actually, it is very interesting to exploit more students' behaviors (e.g., number of attempts or submission patterns) for modeling their knowledge acquisition with considering some psychological traits like "gaming factor" [73].

Third, even if in our simplified scenario, we still argue that there are some possible feasible future directions. On one hand, it is valuable to explore different forms of both learning curve and forgetting curve for tracking student progress and analyze their impacts in practice [2, 3]. On the other hand, all the data we use currently may have some biases inevitably, since we just exploit the performance data of students. However, they may have some offline learning behaviors, e.g., seeking help from instructors or friends, which cannot be recorded. Thus, it is better if we can collect such data to complement the modeling and alleviate this contradiction issue.

Henceforth, we are willing to incorporate other factors, e.g., students' social relationship, for detailed analyzing the learning process. Along this line, some representative and promising methods, such as diffusion analysis [51, 72] and network embedding [13, 66, 67], could be potentially helpful.

Last, in real-world systems, it is also an important issue of how to design the attractive UI visualizations for model results, which can improve the engagement of students [6]. We believe that all above research directions could help the online learning systems.

9 CONCLUSIONS

In this article, we provided a focused study on dynamically diagnosing the knowledge proficiency of students. We designed two explanatory probabilistic matrix factorization models, the *Knowledge Proficiency Tracing* (KPT) model and the *Exercise-correlated Knowledge Proficiency Tracing* (EKPT) model, by incorporating important factors in the learning process of students. To be specific, the KPT associated each exercise with a knowledge vector given the help of Q-matrix and represented each student with a proficiency vector at each time in the same knowledge space. Then, we jointly applied the classical educational learning theories (i.e., *learning curve* and *forgetting curve*) to capture the change of students' knowledge proficiency over time. Furthermore, the improved EKPT model could capture the connectivity relationships among exercises with the same knowledge concepts, benefiting the predictive performance. Finally, we accomplished three practical educational tasks, i.e., knowledge estimation, score prediction, and diagnosis result visualization, based on KPT and EKPT. Extensive experiments on four real-world datasets for these diagnostic tasks confirmed the effectiveness and interpretability of our models. The new models we proposed and related results should significantly benefit the development of online learning systems and related research on educational management. We hope this work will lead to more studies in the future.

ACKNOWLEDGMENTS

The authors thank Prof. Chengxiang Zhai for his constructive advice and the anonymous reviewers for their valuable comments. This article was an expanded version of Reference [11], which appeared in *Proceedings of the 26th ACM International Conference on Information and Knowledge Management (CIKM'17).*

[9] https://www.duolingo.com/.
[10] https://codeforces.com/.

REFERENCES

[1] Ashton Anderson, Daniel Huttenlocher, Jon Kleinberg, and Jure Leskovec. 2014. Engaging with massive online courses. In *Proceedings of the 23rd International Conference on World Wide Web*. ACM, 687–698.

[2] Michel Jose Anzanello and Flavio Sanson Fogliatto. 2011. Learning curve models and applications: Literature review and research directions. *Int. J. Industr. Ergonom.* 41, 5 (2011), 573–583.

[3] Lee Averell and Andrew Heathcote. 2011. The form of the forgetting curve and the fate of memories. *J. Math. Psychol.* 55, 1 (2011), 25–35.

[4] Tiffany Barnes. 2005. The Q-matrix method: Mining student response data for knowledge. In *Proceedings of the American Association for Artificial Intelligence Educational Data Mining Workshop*. 1–8.

[5] Jordan Barria-Pineda, Julio Guerra, Yun Huang, and Peter Brusilovsky. 2017. Concept-level knowledge visualization for supporting self-regulated learning. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces Companion*. ACM, 141–144.

[6] Dror Ben-Naim, Nadine Marcus, and Mike Bain. 2008. Visualization and analysis of student interaction in an adaptive exploratory learning environment. In *Proceedings of the International Workshop on Intelligent Support for Exploratory Environment, EC-TEL*, Vol. 8.

[7] Léon Bottou. 2010. Large-scale machine learning with stochastic gradient descent. In *Proceedings of the International Conference on Computational Statistics (COMPSTAT'10)*. Springer, 177–186.

[8] Hugh Burns, Carol A. Luckhardt, James W. Parlett, and Carol L. Redfield. 2014. *Intelligent Tutoring Systems: Evolutions in Design*. Psychology Press.

[9] Hao Cen, Kenneth Koedinger, and Brian Junker. 2006. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *Intelligent Tutoring Systems*. Springer, 164–175.

[10] Yunxiao Chen, Jingchen Liu, Gongjun Xu, and Zhiliang Ying. 2015. Statistical analysis of Q-matrix-based diagnostic classification models. *J. Amer. Statist. Assoc.* 110, 510 (2015), 850–866.

[11] Yuying Chen, Qi Liu, Zhenya Huang, Le Wu, Enhong Chen, Runze Wu, Yu Su, and Guoping Hu. 2017. Tracking knowledge proficiency of students with educational priors. In *Proceedings of the ACM Conference on Information and Knowledge Management*. ACM, 989–998.

[12] Albert T. Corbett and John R. Anderson. 1994. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Model. User-adapt. Interact.* 4, 4 (1994), 253–278.

[13] Peng Cui, Xiao Wang, Jian Pei, and Wenwu Zhu. 2018. A survey on network embedding. *IEEE Trans. Knowl. Data Eng.* 31, 5 (2018), 833–852.

[14] Thanasis Daradoumis, Roxana Bassi, Fatos Xhafa, and Santi Caballé. 2013. A review on massive e-learning (MOOC) design, delivery and assessment. In *Proceedings of the 8th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC'13)*. IEEE, 208–213.

[15] Jimmy De La Torre. 2009. DINA model and parameter estimation: A didactic. *J. Educat. Behav. Stat.* 34, 1 (2009), 115–130.

[16] Jimmy De La Torre. 2011. The generalized DINA model framework. *Psychometrika* 76, 2 (2011), 179–199.

[17] Louis V. DiBello, Louis A. Roussos, and William Stout. 2006. 31a review of cognitively diagnostic assessment and a summary of psychometric models. *Handbook Stat.* 26 (2006), 979–1030.

[18] Charlotte Dignath and Gerhard Büttner. 2008. Components of fostering self-regulated learning among students. A meta-analysis on intervention studies at primary and secondary school level. *Metacogn. Learn.* 3, 3 (2008), 231–264.

[19] Hermann Ebbinghaus. 2013. Memory: A contribution to experimental psychology. *Ann. Neurosci.* 20, 4 (2013), 155–156.

[20] Susan E. Embretson and Steven P. Reise. 2013. *Item Response Theory*. Psychology Press.

[21] Mi Fei and Dit-Yan Yeung. 2015. Temporal models for predicting student dropout in massive open online courses. In *Proceedings of the IEEE International Conference on Data Mining Workshop (ICDMW'15)*. IEEE, 256–263.

[22] Rana Forsati, Mehrdad Mahdavi, Mehrnoush Shamsfard, and Mohamed Sarwat. 2014. Matrix factorization with explicit trust and distrust side information for improved social recommendation. *ACM Trans. Info. Syst.* 32, 4 (2014), 17.

[23] José González-Brenes, Yun Huang, and Peter Brusilovsky. 2014. General features in knowledge tracing to model multiple subskills, temporal item response theory, and expert knowledge. In *Proceedings of the 7th International Conference on Educational Data Mining*. University of Pittsburgh, 84–91.

[24] Sten Govaerts, Katrien Verbert, Joris Klerkx, and Erik Duval. 2010. Visualizing activities for self-reflection and awareness. In *Proceedings of the International Conference on Web-based Learning*. Springer, 91–100.

[25] Rebecca Grossman and Eduardo Salas. 2011. The transfer of training: What really matters. *Int. J. Train. Dev.* 15, 2 (2011), 103–120.

[26] Xiangnan He, Zhankui He, Jingkuan Song, Zhenguang Liu, Yu-Gang Jiang, and Tat-Seng Chua. 2018. NAIS: Neural attentive item similarity model for recommendation. *IEEE Trans. Knowl. Data Eng.* 30, 12 (2018), 2354–2366.

[27] Liang Hu, Longbing Cao, Jian Cao, Zhiping Gu, Guandong Xu, and Jie Wang. 2017. Improving the Quality of Recommendations for Users and Items in the Tail of Distribution. *ACM Trans. Info. Syst.* 35, 3 (2017), 25.

[28] Zhenya Huang, Qi Liu, Enhong Chen, Hongke Zhao, Mingyong Gao, Si Wei, Yu Su, and Guoping Hu. 2017. Question difficulty prediction for READING problems in standard tests. In *Proceedings of the AAAI Conference on Artificial Intelligence.* 1352–1359.

[29] Zhenya Huang, Qi Liu, Chengxiang Zhai, Yu Yin, Enhong Chen, Weibo Gao, and Guoping Hu. 2019. Exploring multi-objective exercise recommendations in online education systems. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management.* ACM, 1261–1270.

[30] Tanja Käser, Severin Klingler, Alexander G. Schwing, and Markus Gross. 2017. Dynamic Bayesian networks for student modeling. *IEEE Trans. Learn. Technol.* 10, 4 (2017), 450–462.

[31] Mohammad Khajah, Rowan Wing, Robert Lindsey, and Michael Mozer. 2014. Integrating latent-factor and knowledge-tracing models to predict individual differences in learning. In *Proceedings of the Conference on Educational Data Mining.* 99–106.

[32] René F. Kizilcec, Mar Pérez-Sanagustín, and Jorge J. Maldonado. 2017. Self-regulated learning strategies predict learner behavior and goal attainment in massive open online courses. *Comput. Educat.* 104 (2017), 18–33.

[33] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[34] Xin Li, Mingming Jiang, Huiting Hong, and Lejian Liao. 2017. A time-aware personalized point-of-interest recommendation via high-order tensor factorization. *ACM Trans. Info. Syst.* 35, 4 (2017), 31.

[35] Defu Lian, Kai Zheng, Yong Ge, Longbing Cao, Enhong Chen, and Xing Xie. 2018. GeoMF++: Scalable location recommendation via joint geographical modeling and matrix factorization. *ACM Trans. Info. Syst.* 36, 3 (2018), 33.

[36] Chih-Jen Lin. 2007. Projected gradient methods for nonnegative matrix factorization. *Neural Comput.* 19, 10 (2007), 2756–2779.

[37] Jingchen Liu, Gongjun Xu, and Zhiliang Ying. 2012. Data-driven learning of Q-matrix. *Appl. Psychol. Measure.* 36, 7 (2012), 548–564.

[38] Qi Liu, Runze Wu, Enhong Chen, Guandong Xu, Yu Su, Zhigang Chen, and Guoping Hu. 2018. Fuzzy cognitive diagnosis for modelling examinee performance. *ACM Trans. Intell. Syst. Technol.* 9, 4 (2018), 48.

[39] Yuping Liu, Qi Liu, Runze Wu, Enhong Chen, Yu Su, Zhigang Chen, and Guoping Hu. 2016. Collaborative learning team formation: A cognitive modeling perspective. In *Proceedings of the International Conference on Database Systems for Advanced Applications.* Springer, 383–400.

[40] Wei Lu, Fu-Lai Chung, Wenhao Jiang, Martin Ester, and Wei Liu. 2018. A deep Bayesian tensor-based system for video recommendation. *ACM Trans. Info. Syst.* 37, 1 (2018), 7.

[41] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *J. Mach. Learn. Res.* 9 (Nov. 2008), 2579–2605.

[42] Jiaxin Mao, Yiqun Liu, Noriko Kando, Min Zhang, and Shaoping Ma. 2018. How does domain expertise affect users' search interaction and outcome in exploratory search? *ACM Trans. Info. Syst.* 36, 4 (2018), 42.

[43] Sein Minn, Yi Yu, Michel C. Desmarais, Feida Zhu, and Jill-Jenn Vie. 2018. Deep knowledge tracing and dynamic student classification for knowledge tracing. In *Proceedings of the IEEE International Conference on Data Mining (ICDM'18).* IEEE, 1182–1187.

[44] Andriy Mnih and Ruslan R. Salakhutdinov. 2008. Probabilistic matrix factorization. In *Advances in Neural Information Processing Systems.* MIT Press, 1257–1264.

[45] David A. Nembhard and Mustafa V. Uzumeri. 2000. An individual-based description of learning within an organization. *IEEE Trans. Eng. Manage.* 47, 3 (2000), 370–378.

[46] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Jingfang Xu, and Xueqi Cheng. 2017. Deeprank: A new deep architecture for relevance ranking in information retrieval. In *Proceedings of the ACM Conference on Information and Knowledge Management.* ACM, 257–266.

[47] Aditya Parameswaran, Petros Venetis, and Hector Garcia-Molina. 2011. Recommendation systems with complex constraints: A course recommendation perspective. *ACM Trans. Info. Syst.* 29, 4 (2011), 20.

[48] Zachary A. Pardos and Neil T. Heffernan. 2011. KT-IDEM: Introducing item difficulty to the knowledge tracing model. In *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization.* Springer, 243–254.

[49] Scott G. Paris and Alison H. Paris. 2001. Classroom applications of research on self-regulated learning. *Educat. Psychol.* 36, 2 (2001), 89–101.

[50] Philip I. Pavlik Jr., Hao Cen, and Kenneth R. Koedinger. 2009. Performance factors analysis—A new alternative to knowledge tracing. In *Proceedings of the 2009 conference on Artificial Intelligence in Education: Building Learning Systems that Care: From Knowledge Representation to Affective Modelling.* IOS Press, 531–538.

[51] Hongbin Pei, Bo Yang, Jiming Liu, and Lei Dong. 2018. Group sparse Bayesian learning for active surveillance on epidemic dynamics. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence.* 800–807.

[52] Chris Piech, Jonathan Bassen, Jonathan Huang, Surya Ganguli, Mehran Sahami, Leonidas J. Guibas, and Jascha Sohl-Dickstein. 2015. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*. MIT Press, 505–513.

[53] Q. Liu, Z. Huang, Y. Yin, E. Chen, H. Xiong, Y. Su, and G. Hu. 2019. EKT: Exercise-aware knowledge tracing for student performance prediction. *IEEE Trans. Knowl. Data Eng.* (2019), 1–1. DOI: https://doi.org/10.1109/TKDE.2019.2924374

[54] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*. AUAI Press, 452–461.

[55] Jeff W. Rickel. 1989. Intelligent computer-aided instruction: A survey organized around system components. *IEEE Trans. Syst. Man Cybernet.* 19, 1 (1989), 40–57.

[56] Cristóbal Romero and Sebastián Ventura. 2010. Educational data mining: A review of the state of the art. *IEEE Trans. Syst. Man Cybernet. Part C (Appl. Rev.)* 40, 6 (2010), 601–618.

[57] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*. ACM, 285–295.

[58] B. Settles, C. Brust, E. Gustafson, M. Hagiwara, and N. Madnani. 2018. Second language acquisition modeling. In *Proceedings of the NAACL-HLT Workshop on Innovative Use of NLP for Building Educational Applications (BEA'18)*. ACL.

[59] 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.

[60] Yuan Sun, Shiwei Ye, Shunya Inoue, and Yi Sun. 2014. Alternating recursive method for Q-matrix learning. In *Proceedings of the 7th International Conference on Educational Data Mining*. 14–20.

[61] Kikumi K. Tatsuoka. 1983. Rule space: An approach for dealing with misconceptions based on item response theory. *J. Educat. Measure.* 20, 4 (1983), 345–354.

[62] Nguyen Thai-Nghe, Lucas Drumond, Artus Krohn-Grimberghe, and Lars Schmidt-Thieme. 2010. Recommender system for predicting student performance. *Procedia Comput. Sci.* 1, 2 (2010), 2811–2819.

[63] Nguyen Thai-Nghe, Tomáš Horváth, and Lars Schmidt-Thieme. 2011. Factorization models for forecasting student performance. In *Proceedings of the 4th International Conference on Educational Data Mining (EDM'11)*. 11.

[64] Khushboo Thaker, Yun Huang, Peter Brusilovsky, and He Daqing. 2018. Dynamic knowledge modeling with heterogeneous activities for adaptive textbooks. In *Proceedings of the 11th International Conference on Educational Data Mining*. 592–595.

[65] Fei Wang, Qi Liu, Enhong Chen, and Zhenya Huang. 2019. Interpretable Cognitive Diagnosis with Neural Network. arxiv:cs.LG/1908.08733

[66] Hao Wang, Enhong Chen, Qi Liu, Tong Xu, Dongfang Du, Wen Su, and Xiaopeng Zhang. 2018. A united approach to learning sparse attributed network embedding. In *Proceedings of the IEEE International Conference on Data Mining (ICDM'18)*. IEEE, 557–566.

[67] Hao Wang, Tong Xu, Qi Liu, Defu Lian, Enhong Chen, Dongfang Du, Han Wu, and Wen Su. 2019. MCNE: An end-to-end framework for learning multiple conditional network representations of social network. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 1064–1072.

[68] Minhong Wang, Jun Peng, Bo Cheng, Hance Zhou, and Jie Liu. 2011. Knowledge visualization for self-regulated learning. *J. Educat. Technol. Soc.* 14, 3 (2011), 28–42.

[69] Tzu-Hua Wang. 2011. Developing Web-based assessment strategies for facilitating junior high school students to perform self-regulated learning in an e-Learning environment. *Comput. Educat.* 57, 2 (2011), 1801–1812.

[70] Xiaojing Wang, James O. Berger, Donald S. Burdick, et al. 2013. Bayesian analysis of dynamic item response models in educational testing. *Ann. Appl. Stat.* 7, 1 (2013), 126–153.

[71] Le Wu, Yong Ge, Qi Liu, Enhong Chen, Richang Hong, Junping Du, and Meng Wang. 2017. Modeling the evolution of users' preferences and social links in social networking services. *IEEE Trans. Knowl. Data Eng.* 29, 6 (2017), 1240–1253.

[72] Le Wu, Peijie Sun, Yanjie Fu, Richang Hong, Xiting Wang, and Meng Wang. 2019. A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 235–244.

[73] Runze Wu, Guandong Xu, Enhong Chen, Qi Liu, and Wan Ng. 2017. Knowledge or gaming?: Cognitive modelling based on multiple-attempt response. In *Proceedings of the 26th International Conference on World Wide Web Companion*. International World Wide Web Conferences Steering Committee, 321–329.

[74] Masoud Yazdani. 1986. Intelligent tutoring systems survey. *Artific. Intell. Rev.* 1, 1 (1986), 43–52.

[75] Michael V. Yudelson, Kenneth R. Koedinger, and Geoffrey J. Gordon. 2013. Individualized Bayesian knowledge tracing models. In *Proceedings of the International Conference on Artificial Intelligence in Education*. Springer, 171–180.

[76] Hanwang Zhang, Fumin Shen, Wei Liu, Xiangnan He, Huanbo Luan, and Tat-Seng Chua. 2016. Discrete collaborative filtering. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval.* ACM, 325–334.

[77] Jiani Zhang, Xingjian Shi, Irwin King, and Dit-Yan Yeung. 2017. Dynamic key-value memory networks for knowledge tracing. In *Proceedings of the 26th International Conference on World Wide Web.* International World Wide Web Conferences Steering Committee, 765–774.

[78] 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 Trans. Info. Syst.* 36, 3 (2018), 27.

[79] Barry J. Zimmerman. 1990. Self-regulated learning and academic achievement: An overview. *Educat. Psychol.* 25, 1 (1990), 3–17.

Received February 2019; revised December 2019; accepted January 2020