Knowledge-Based Systems 309 (2025) 112740

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Enhancing learning process modeling for session-aware knowledge tracing

Chunli Huang [a], Wenjun Jiang [a,*], Kenli Li [a], Jie Wu [b], Ji Zhang [c]

[a] College of Computer Science and Electronic Engineering, Hunan University, 116 Lu Shan South Road, Changsha, 410082, Hunan, China
[b] Department of Computer and Information Science, Temple University, Philadelphia, 19122, PA, USA
[c] Department of Mathematics and Computing, University of Southern Queensland, Brisbane QLD 4350, Philadelphia, 310012, Queensland, Australia

ARTICLE INFO

Keywords:
Knowledge tracing
Learning process modeling
Fine-grained learning behavior
Knowledge state shifts

ABSTRACT

Session-aware knowledge tracing tries to predict learners' performance, by splitting learners' sequences into sessions and modeling their learning within and between sessions. However, there still is a lack of comprehensive understanding of the learning processes and session-form learning patterns. Moreover, the knowledge state shifts between sessions at the knowledge concept level remain unexplored. To this end, we conduct in-depth data analysis to understand learners' learning processes and session-form learning patterns. Then, we perform an empirical study validating knowledge state shifts at the knowledge concept level in real-world educational datasets. Subsequently, a method of Enhancing Learning Process Modeling for Session-aware Knowledge Tracing, ELPKT, is proposed to capture the knowledge state shifts at the knowledge concept level and track knowledge state across sessions. Specifically, the ELPKT models learners' learning process as intra-sessions and inter-sessions from the knowledge concept level. In intra-sessions, fine-grained behaviors are used to capture learners' short-term knowledge states accurately. In inter-sessions, learners' knowledge retentions and decays are modeled to capture the knowledge state shift between sessions. Extensive experiments on four real-world datasets demonstrate that ELPKT outperforms the existing methods in learners' performance prediction. Additionally, ELPKT shows its ability to capture the knowledge state shifts between sessions and provide interpretability for the predicted results.

1. Introduction

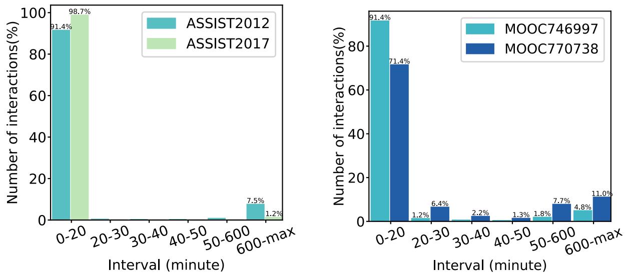
Knowledge tracing (KT) is a fundamental and critical task in intelligent educational technology, which has already been broadly applied in numerous educational scenarios [1]. It aims to assess learners' knowledge proficiency based on their historical learning sequences and predict their performance in future exercises. The learners' performance predicted by KT technology provides timely feedback on their knowledge levels and helps optimize their next learning plans. There are many representative works in the literature, especially deep learning-based KT works, which have achieved state-of-the-art results on most KT benchmark datasets [2]. In existing KT works, most consider various factors related to learners' performance, e.g., questions [3–6], individual differences [4,7,8], temporal effects [5,9–13], and fine-grained behaviors [13], to improve models' prediction ability. Furthermore, the latest studies [14,15] focus on the accuracy of learners' knowledge states generated by KT models. While the above efforts drive progress in knowledge tracing, they all treat learners' interaction sequences as continuous sequences, ignoring that interactions within learners' sequences are non-uniformly distributed. That is, the time intervals between adjacent interactions within learners' sequences vary greatly. To illustrate this phenomenon clearly, as shown in Fig. 1, we statistically analyze the distribution of time intervals between adjacent interactions in four educational datasets (please refer to Section 4.1.1 for the details of datasets).

As seen from Figs. 1(a) and 1(b), the time intervals for most adjacent interactions are small, and only a few of them are larger. For example, the 91.4% of time intervals in the ASSIST2012 dataset are less than 20 min. The phenomenon implies that: (1) learners' online learning is in the form of sessions, where they answer several questions continuously. (2) The time intervals of adjacent interactions within sessions are small, and the time intervals between sessions are larger. As such, learners' session-form learning patterns should be considered to trace their knowledge state in the learning process.

Recently, researchers have focused on the above issues and proposed session-aware KT methods [16,17], introducing a new paradigm for KT. The session-aware knowledge tracing predicts learners' performance with the hierarchical structure and relationship between learners' sessions. Specifically, they split learners' sequences into sessions and model two sequences: interactions within sessions and sequences of different sessions.

[*] Corresponding author.
E-mail addresses: huangchunli@hnu.edu.cn (C. Huang), jiangwenjun@hnu.edu.cn (W. Jiang), lkl@hnu.edu.cn (K. Li), jiewu@temple.edu (J. Wu), ji.zhang@usq.edu.au (J. Zhang).

https://doi.org/10.1016/j.knosys.2024.112740
Received 27 August 2024; Received in revised form 7 November 2024; Accepted 12 November 2024
Available online 21 November 2024



(a) ASSIST2012 and ASSIST2017 (b) The two datasets about MOOC.

Fig. 1. The interval statistics in the four real-world datasets.

While they have achieved promising results, there are still some open challenges in existing session-aware KT works: (1) There is a lack of deep analysis and understanding of learners' learning process and their session-form learning patterns. (2) When considering session-form learning patterns to trace learners' knowledge state, the knowledge state shifts at the knowledge concepts level remain unexplored.

To address the above challenges, we analyze learners' learning processes and their session-form learning patterns from several aspects. Subsequently, we explore the knowledge state shifts between sessions at the knowledge concept level, which aims to uncover that learners' responses to the same knowledge may differ between sessions resulting from too large time intervals. We further propose a method of Enhancing Learning Process Modeling for Session-aware Knowledge Tracing, ELPKT, which aims to capture the knowledge state shifts from the knowledge concepts level and track learners' knowledge state across sessions. The main contributions of this article are summarized as follows:

(1) We conduct in-depth data analysis to understand learners' learning processes and their session-form learning patterns. Furthermore, we undertake an empirical study to validate the knowledge state shifts between sessions from the knowledge concepts level in four real-world educational datasets (Section 4).

(2) We propose Enhancing Learning Process Modeling for Session-aware Knowledge Tracing, ELPKT, which models the learning process as intra- and inter-session from the knowledge concept level. In intra-session, learners' fine-grained behaviors within sessions are used to capture their short-term knowledge states accurately. In inter-session, the knowledge retentions and decays are modeled to capture the knowledge state shift between sessions (Section 5).

(3) We conduct extensive experiments on four real-world datasets, comparing the proposed ELPKT to nine advanced baselines. The experimental results demonstrate that ELPKT outperforms the existing methods in predicting learners' performance. Moreover, they also validate that ELPKT can capture the knowledge state shifts between sessions and provide interpretability for the predicted results (Section 6).

The article is organized as follows: Section 2 reviews related work. Sections 3 and 4 describe the problem formulation and data analysis, supporting the design of our model. In Section 5, the proposed ELPKT model is introduced in detail. Section 6 evaluates the proposed model through extensive experiments and discusses the experiment results. Finally, Section 7 concludes this article.

2. Related work

This section reviews the literature on knowledge tracing (KT) to highlight the urgent issues about KT and our research motivations. In this article, we mainly focus on the knowledge state shifts resulting from temporal effects in KT, hence we discuss two categories of KT works (i.e., KT works not considering temporal effects and KT works considering temporal effects) and our differences.

2.1. KT works not considering temporal effects

These KT models employ various techniques and learning-related factors to trace learners' knowledge state. For instance, DKT [18] first utilizes RNNs to model learners' sequences, and uses the hidden state to represent overall knowledge levels. In subsequent works, DKT+ [19] addresses the reconstruction error and the waveform transition in the DKT. DKVMN [20] and SKVMN [21] have built on DKT by enhancing the knowledge concepts modeling with memory matrices. In SAKT [22], the self-attention mechanism is integrated to address the shortcomings of DKT and DKVMN. The above classic methods consider knowledge concepts and responses in learners' sequences for knowledge tracing modeling. Some works gradually consider the features besides knowledge concepts to model knowledge state evolution. For example, Zhang et al. [23] improve the DKT by incorporating more features (e.g., exercise tags, response times). Other models like IEKT [7], DIMKT [3], and interpretable KT models [4,6] focus on evaluating cognitive abilities, exercise difficulty, and individual skill mastery for better knowledge level assessment. Additionally, Recent studies [14,15] focus on the accuracy of learners' knowledge states generated by KT models and explore stable knowledge tracing.

Most KT works [3,14,18–21] are RNN-based, with a lower calculation cost due to their fewer parameters. Because the gradient vanishes or explodes in RNN models, RNN-based KT models generally model the short learners' sequences truncated by long sequences, which may cause them to ignore the temporal effects in tracing dynamic knowledge states. Some recent works [15,24,25] introduce Transformers, which be applied efficiently in long-sequence modeling. However, the large number of parameters in Transformer-based models increases the calculation cost. Considering that there are fewer interactions within sessions and RNN is a common KT model structure with the advantage of fewer parameters, the proposed ELPKT model still selects RNN as the base model to track learners' knowledge state on each knowledge concept.

2.2. KT works considering temporal effects

Education psychologists [26–28] proposed that the temporal factor may lead to learners' knowledge forgetting. Therefore, some works consider temporal effects on tracing knowledge state in a unified way. Few other studies consider learners' session-form learning patterns and suggest that large time intervals may cause knowledge state shifts compared to small time intervals. Then, we further categorized the former as KT works modeling temporal effects in a unified way and the latter as session-aware KT works.

2.2.1. KT works modeling temporal effects in a unified way

These works mainly model temporal effects on knowledge forgetting and the correlation between interactions in a unified way to trace the knowledge state.

Modeling temporal effects on learners' knowledge forgetting. In these studies, researchers address the phenomenon of knowledge forgetting in educational contexts through various innovative approaches. They leverage forgetting factors [29], time-based features [9,11,30,31] to capture and understand knowledge forgetting in learning processes. Studies, such as LPKT [11] and LFKT [32], consider how the time intervals between interactions impact both learning gain and forgetting. Im et al. [33] propose FoLiBi, a model reflecting forgetting behaviors in linear bias. Abdelrahman and Wang [31] explore two critical forgetting features and integrate forget gating mechanism into attention memory structure to capture forgetting. Additionally, LBKT [13] models learners' forgetting by combining the time interval and fine-grained learning behavior.

Modeling temporal effects on correlations between interactions. These works simulate the decay of the correlation between interactions over time, which is implicit forgetting modeling. In these works, researchers explore diverse methods to understand and model temporal dynamics

Table 1
Notations and Descriptions.

Notations	Descriptions
U, E_1	The set of learners and exercises.
C	The set of knowledge concepts (KCs).
LP	The complete learning process.
O	The time interval between adjacent sessions.
$\boldsymbol{E}_1$	The embedding matrix of exercises.
$\boldsymbol{E}_2$	The embedding matrix of correctness labels.
$\boldsymbol{D}$	The embedding matrix of exercises' difficulty.
e, $\boldsymbol{e}$	The exercise and its embedding.
r, $\boldsymbol{r}$	The correctness label and its embedding.
$\boldsymbol{A}_t$	The embedding matrix of answer times.
$\boldsymbol{A}_c$	The embedding matrix of attempt count.
$\boldsymbol{H}_c$	The embedding matrix of hint count.
at, $\boldsymbol{at}$	The answer time and its embedding.
ac, $\boldsymbol{ac}$	The attempt count and its embedding.
hc, $\boldsymbol{hc}$	The hint count and its embedding.
Q	The relation matrix of KCs and exercises.
$\boldsymbol{q}$	The knowledge concept vector of exercise.
d, $\boldsymbol{d}$	The difficulty level and its embedding.
$\boldsymbol{L}$	The learning interaction embedding.
$\boldsymbol{B}$	The fine-grained behavior embedding.
$\boldsymbol{H}^S$, $\boldsymbol{H}^L$	The short-term and long-term knowledge states.
$\boldsymbol{F}_q$	The vector of knowledge practice frequency.

in educational data, including cross-effects decay [12,34], the diminishing importance of interactions [5,10,35,36], and the impact of recent exercises [37], to enhance predictive accuracy and insights into learners' performance in their learning process.

The above works propose several representative methods of modeling temporal effects to trace learners' knowledge states. However, they neglect the learners' session-form learning patterns and the knowledge state shift resulting from large time intervals, which may lead to KT models failing to track learners' knowledge states promptly.

2.2.2. Session-aware knowledge tracing

Researchers focus on the knowledge state shifts and propose session-aware knowledge tracing. Ke et al. [17] introduce HiTSKT, a method that splits sessions when the time interval between interactions exceeds 10 h. They use an interaction encoder that captures the relationship between interactions to model intra-session, and a session encoder that captures the knowledge state shift between sessions from the overall knowledge level to model inter-session. Shen et al. [16] introduce QKT, which splits quizzes by quiz ID. Then, they employ the RNN variant that captures the knowledge relationship between interactions to model intra-quiz and combine the RNN variant and self-attentive encoder to capture the knowledge state shift between quizzes from the overall knowledge level and model inter-quiz.

While the above works focused on learners' knowledge shifts between sessions, there still is a lack of deep analysis and understanding of the learning process and session-form learning patterns. In addition, the knowledge state shifts between sessions at the fine-grained knowledge concepts level remain unexplored.

Our Differences. To address these issues, we first conduct a comprehensive data analysis to understand learners' learning process and session-form learning patterns. Then, we explore the knowledge state shifts between sessions from the knowledge concepts level through empirical study. To effectively model learners' complete learning process, we consider the fine-grained learning behaviors to model intra-session and capture the knowledge state shifts between sessions from the knowledge concepts level to model inter-session.

3. Problem formulation

This section first explains the terms used in this article. Then, the studied KT problem is formulated. For clarity, the notations utilized in this article are summarized in Table 1.

3.1. Term definition

Learning Session. A learning session S is defined as a continuous interaction sequence online. Considering the time intervals between adjacent interactions vary in learners' sequences, we split learners' sequences into sessions. The method of splitting sessions is introduced in Section 4.3.

Offline Time. Offline time $O_{p-1,p}$ denotes the time intervals between the adjacent $p-1$th and pth session, where the learning interactions are unobservable.

Learning Process. Based on learners' session-form learning pattern, the learning process is defined as consisting of online intra-sessions and offline inter-sessions. A complete learning process LP is represented as $LP = \{S^1, O_{1,2}, S^2, \ldots, S^{p-1}, O_{p-1,p}, S^p, O_{p,p+1}\}$, where S^p denotes all interactions in the pth session and $O_{p,p+1}$ denotes the offline time between the pth and $p+1$th session.

3.2. Problem definition

In an online learning system, there are I learners and J exercises covering M knowledge concepts (KCs), and learner $u_i \in U$, exercise $e_j \in E_1$ and the knowledge concept $c_m \in C$. The relation between exercises and KCs is generally represented by a matrix Q, where $Q \in \mathbb{R}^{J\times M}$. The jth row of Q is the knowledge concept vector about exercise e_j. Each element of Q is either 0 or 1, indicating whether exercise e_j contains KC c_m ($Q_{jm} = 1$) or not ($Q_{jm} = 0$).

A learner's interactions in the pth session is denoted as $S^p = \{s_1^p, s_2^p, \ldots, s_t^p\}$, where $s_t^p = (e_t^p, r_t^p, b_t^p)$ is the learner's interaction at time step t in this session. e_t^p is the exercise, and r_t^p is the correctness label (1 for correct, 0 for incorrect). $b_t^p = (at_t^p, ac_t^p, hc_t^p)$ is the learner's fine-grained behaviors about solving e_t^p, where at_t^p is the answer time, ac_t^p is the attempt count and hc_t^p is the hint count.

We define the enhancing learning process modeling for session-aware knowledge tracing problem as follows: Given a learner's learning process, $LP = \{S^1, O_{1,2}, S^2, \ldots, S^{p-1}, O_{p-1,p}, S^p, O_{p,p+1}\}$, where $S^p = [(e_1^p, r_1^p, b_1^p), \ldots, (e_t^p, r_t^p, b_t^p)]$, and $O_{p,p+1}$ is the offline time between the pth and $p+1$th session, our goal is to predict learners' performance in the next $p+1$ session, by monitoring learners' dynamic knowledge states on each knowledge concept in the pth session and capture the knowledge state shifts in offline time $O_{p,p+1}$.

4. Data analysis and processing

This section first introduces the preliminaries: the datasets and the correlation analysis method. Then, we conduct comprehensive data analysis to understand learners' learning processes and session-form learning patterns. Additionally, we split sessions and preprocess data based on the data analysis. Finally, we perform an empirical study to validate the knowledge state shifts between sessions from the knowledge concept level in real-world education datasets, which motivates our model design.

4.1. Preliminary

4.1.1. Dataset

The datasets used in our data analysis and experiments are from the Intelligent Tutoring Systems (e.g., ASSIST2012 and ASSIST2017) and MOOC platforms (e.g., MOOC746997 and MOOC770738).

ASSIST2012[1] **and ASSIST2017**[2] **dataset**. The two datasets are derived from the online tutoring system ASSISTments.[3] They record the

[1] https://sites.google.com/site/ASSISTdata/home/2012-13-school-data-with-affect

[2] https://sites.google.com/view/ASSISTdatamining/dataset

[3] https://new.assistments.org/

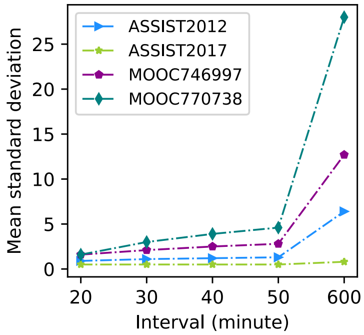


Fig. 2. Interactions uniformity.

interaction data of learners' answering math exercises during the 2012–2013 and 2004–2007 academic years, respectively, in ASSISTments. They both include learners' IDs, exercises, skills, and learning behavior, i.e., answer time, hint counts, and attempt counts.

MOOC746997 and MOOC770738.[4] They are two course learning datasets from MOOCRadar [38]. The raw data comes from XuetangX,[5] which is a well-known MOOCs platform. The two datasets respectively record the learners' interactions in the courses "Fundamentals of Analog Electronics" and "Data Structure" in 2020. They include learners' IDs, exercises, knowledge concepts, and behavior, e.g., submit time and submit count.

4.1.2. Correlation analysis method

Conditional mutual information (CMI) is introduced to quantify the correlation of learners' responses to the same KCs and analyze the knowledge state shifts between sessions. The CMI represents the degree of correlation between two random variables under a given restrictive condition [12]. The larger the CMI, the higher the correlation between the two random variables. It can be used to quantify the correlation of learners' responses to the same KCs within and between sessions. A lower correlation of responses to the same KCs suggests potential knowledge state shifts. Therefore, the knowledge state shifts between sessions can be uncovered by utilizing CMI.

Given a specific condition c, we first identify all interaction pairs (x_i, x_j) in the learners' sequence that satisfies the condition c. Then, we treat the interaction pairs' responses (r_i, r_j) as random variables. The conditional mutual information (CMI) is calculated as follows:

$$CMI(r_i; r_j) = \sum_{r_i \in \{0,1\}} \sum_{r_j \in \{0,1\}} P(r_i, r_j) \log \frac{P(r_i, r_j)}{P(r_i)P(r_j)} \tag{1}$$

for the CMI of learners' responses to the same KCs within sessions (or between sessions), the restrictive condition c can be considered as adjacent interactions involving the same KCs within sessions (or interaction pairs involving the same KCs between sessions). The joint probability $P(r_i, r_j)$ and marginal probability $P(r_i)$, $P(r_j)$ can be obtained by calculating the occurrence frequency of interaction pairs that satisfy the restrictive condition c.

4.2. Data analysis

To deeply understand the learning processes and session-form learning patterns, we undertake an in-depth data analysis. Learners' learning processes contain multiple sessions and offline time between sessions. Considering the distribution of time intervals in Fig. 1 and there are lacking session boundary identifiers in the datasets, we design a set of thresholds θ ($\theta \in \{20, 30, 40, 50, 600\}$) to split sessions in four datasets. Our analysis delves into three key aspects of the sessions and learning processes: interaction uniformity, session duration, and offline time.

4.2.1. Interactions uniformity

To analyze the interaction uniformity within sessions, referring to [39], we calculate the mean standard deviation of intervals between adjacent interactions within sessions, as shown in Fig. 2. The smaller mean standard deviation indicates a more uniform interaction distribution within sessions. From Fig. 2, we can observe that **sessions split by different interval thresholds exhibit different interactions uniformity. Moreover, larger interval thresholds result in decreased interaction uniformity**.

4.2.2. Session duration and offline time statistics

To explore learners' time allocation in online and offline learning, we analyze the distribution of session duration and offline time under sessions split by different thresholds, as shown in Fig. 3. Session duration refers to the continuous learning duration in a session, and offline time denotes the time interval between adjacent sessions. We perform log transformations on session duration and offline time. In Fig. 3, we have the following findings:

Learners tend to spend a short time on continuous online learning. As seen in Fig. 3, regardless of the time interval threshold, the durations of most sessions are around 40 min in all datasets. For example, in ASSIST2012, the durations of 75% sessions (between the minimum and Q3 of boxes) are less than 2^5 (i.e., 32) minutes. It suggests that most learners spend less time learning online continuously. However, by carefully analyzing four datasets, we observed that sessions divided by too large intervals (e.g., 600 min) exhibit over 10% exceptions with extended durations, which does not align with learners' short online sessions.

Learners' offline time is much longer than their session duration. In Fig. 3, regardless of the threshold, the time in 75% of offline (between the Q1 and maximum of boxes) is greater than 20 h in ASSIST2012, and it is even greater than eight days in ASSIST2017. When smaller interval thresholds split sessions, the time in 50% of offlines (between the minimum and median) is smaller than 20 h in MOOC746997, and the time in 75% of offline is smaller than 20 h in MOOC770738. This indicates that learners have substantial offline periods where learning activities are not recorded. Additionally, *learners' offline time on the MOOC platform is shorter than in the online tutoring system (ASSISTments)*. This phenomenon could be attributed to frequent updates of course materials on MOOC platforms, which encourage learners to engage regularly to keep their learning progress. In contrast, the ASSISTments may focus more on personalized learning guidance than learning resource updates.

Sessions splitting by too large time intervals results in longer session duration and offline time. This phenomenon may lead to two issues: (1) *It blurs the boundary between online and offline learning, as learners may engage in offline activities within sessions, complicating intra-session modeling.* (2) *Too long offline periods between sessions may hinder the model's ability to capture timely shifts in learners' knowledge states*.

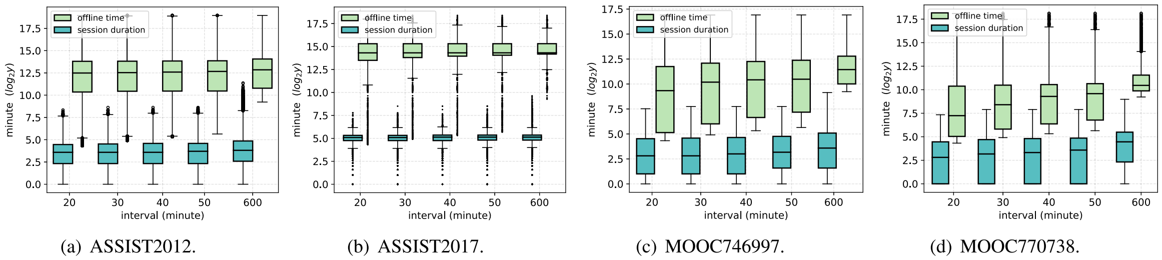
4.3. Data processing

We first set an interval threshold to split sessions based on the observations from the above data analysis, i.e., sessions split by too large thresholds may lead to non-uniform interactions within sessions and too long offline periods between sessions. Studies on learning and attention [40] indicate that the human attention span for learning ranges from 20 to 40 min. Considering the learning process analysis and practical educational applications, we target thresholds that make the duration of over 90% of sessions after splitting to match learners' short online learning patterns and human attention span for learning. As such, we set 30 min as the default interval threshold to split sessions. We also test the performance of our model that runs in sessions split by different interval thresholds in Section 6.2.4.

For splitting sessions, we first remove records containing missing fields and sort the interactions in ascending chronological order in four

[4] https://github.com/THU-KEG/MOOC-Radar

[5] https://www.xuetangx.com/



(a) ASSIST2012. (b) ASSIST2017. (c) MOOC746997. (d) MOOC770738.

Fig. 3. The session duration and offline time distribution under different interval thresholds. In the figure, each bar in the boxes from bottom to top is the minimum, first quartile (Q1), median, third quartile (Q3), and maximum of data, respectively. Besides, it also shows the outliers that are less than the minimum or greater than the maximum.

Table 2
Statistics of all datasets.

DataSet	ASSIST2012	ASSIST2017	MOOC746997	MOOC770738
learners	20 200	1709	1022	873
Exercises	52 624	3162	550	99
Knowledge concepts	265	102	265	80
Interactions	2,600,869	942,816	97,218	46,369
Sessions	233,757	13,873	8102	10,788
Interactions per learners *	31/ 70/ 152	239/ 443/ 743	22/ 47/ 132	25/ 56/ 87
Sessions per learners*	3/ 6/ 14	5/ 8/ 10	2/ 4/ 10	5/ 11/ 19
Interactions per session *	4/ 8/ 14	31/ 53/ 86	5/ 8/ 14	2/ 3/ 5

∗ indicates the data property's first quartile, median, and third quartile, respectively.

datasets. Then, based on the default interval threshold θ (i.e., 30 min), we split learners' sequences into sessions. Specifically, if the time interval between two adjacent interactions is larger than θ, we divide them into two adjacent sessions. For all datasets, we remove the sessions with fewer than two interactions and ensure learners have at least two sessions to guarantee enough sessions for intra- and inter-session modeling. The details of the processed datasets are shown in Table 2.

4.4. Empirical study: Knowledge states shifts between sessions

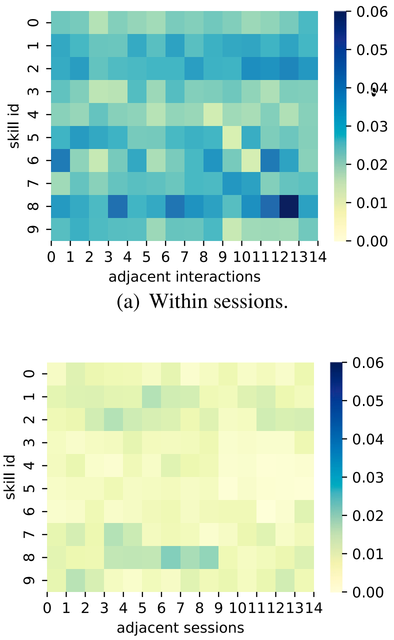
To illustrate the effect of large time intervals on learners' knowledge states, we further validate the knowledge state shifts between sessions from the knowledge concept level in real-world education datasets through empirical study. According to the processed datasets and correlation analysis method in Section 4.1.2, we separately calculate the conditional mutual information (CMI) of learners' responses to the same KCs within and between sessions.

We present the CMI of learners' responses to the same KCs within and between sessions on the top-10 KCs with the highest frequency in the ASSIST2012, as shown in Figs. 4(a) and 4(b). In Fig. 4, the skill id represents the top-10 KCs in the ASSIST2012. Each cell is the CMI of learners' responses to the same KCs. The darker the color of the cell, the larger the CMI. The lower CMI indicates a lower correlation of responses to the same KCs, which implies potential knowledge state shifts. By comparing and analyzing Figs. 4(a) and 4(b), we draw the following important findings:

The knowledge state evolves dynamically within and between sessions. It suggests that both the interactions within sessions and the offline time between adjacent sessions can result in the knowledge state evolving constantly.

The knowledge state shifts are more likely to occur between sessions. As can be found in Figs. 4(a) and 4(b), the CMI of learners' responses to the same KCs between sessions is always lower than that within sessions. It illustrates that learners' performance on the same knowledge concepts tends to be different in adjacent sessions, which denotes the knowledge state shifts between sessions.

The findings validate the existence of knowledge state shifts between sessions in real-world educational datasets. It further motivates us to capture the knowledge state shifts between sessions by inter-session modeling.



(a) Within sessions.

(b) Between sessions.

Fig. 4. The CMI of learners' responses to the same KCs within and between sessions.

5. Model

Based on the processed data and findings in Section 4, this section models the learning process as multiple intra- and inter-session from the knowledge concept level. First, the embedding method is described. Then, in intra-session modeling, learners' fine-grained behaviors within sessions are used to capture the short-term knowledge state accurately. In inter-sessions, the knowledge retentions and decays are modeled to explicitly capture the knowledge state shift between sessions. With intra- and inter-session modeling, ELPKT tracks learners' knowledge state during the entire learning process. The architecture of the proposed ELPKT model is depicted in Fig. 5.

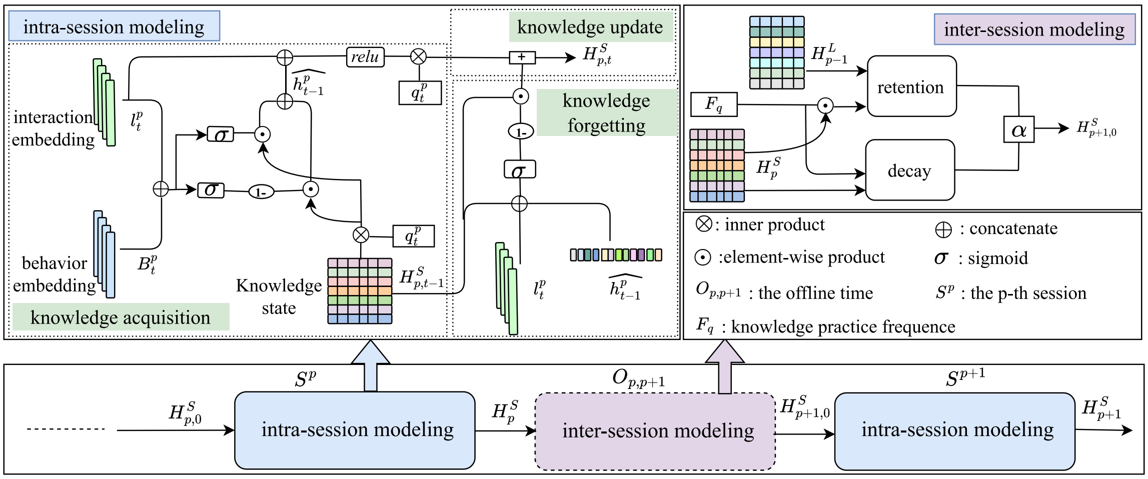


Fig. 5. Overview of the ELPKT model. It contains multiple intra-session modeling and inter-session modeling. In intra-session modeling, learners' short-term knowledge is updated at each time step within sessions. In the figure, we show the process of updating learners' short-term knowledge at tth time step within the pth session in intra-session modeling. Once the pth session finishes, the short-term knowledge state H^S_p is the input of inter-session modeling. After modeling inter-session in $O_{p,p+1}$, the knowledge state $H^S_{p+1,0}$ is the initial state of the $p+1$th session.

5.1. Embedding

We consider the learning-related factors, such as exercises, knowledge concepts, and the learners' behaviors (i.e., answer times, hint counts, and attempt counts). To better understand the proposed ELPKT model, we briefly introduce the embeddings of the following elements.

Exercise Embedding. The exercise embedding matrix $\boldsymbol{E_1}$ store all the exercises embeddings in the dataset, where $\boldsymbol{E_1} \in \mathbb{R}^{J \times d_k}$. J denotes the number of exercises in the dataset, and d_k represents its dimension. Exercise e^p_t that a learner solves at the tth time step in the pth session can be represented as a vector $\boldsymbol{e}^p_t \in \mathbb{R}^{d_k}$.

Response Embedding. The matrix $\boldsymbol{E_2} \in \mathbb{R}^{2 \times d_r}$ encodes the two responses (correct or incorrect), d_r denotes its dimension. $\boldsymbol{r}^p_t \in \mathbb{R}^{d_r}$ is the response vector about e^p_t.

The Knowledge Concept Vector. We use the matrix Q to record the relation between exercises and knowledge concepts (KCs). $Q \in \mathbb{R}^{J \times M}$, where M is the number of KCs in the dataset. The knowledge concept vector of exercise e^p_t can be represented as $\boldsymbol{q}^p_t \in \mathbb{R}^M$.

Exercise Difficulty Embedding. Considering that exercise difficulty affects learners' knowledge mastery and is unlabeled, we calculate the exercises' error rates $\phi(e)$ and map them to 10 levels to denote the exercise difficulty $d(e)$. A higher error rate indicates greater difficulty. The error rate of an exercise $\phi(e) = \frac{\sum_i^{N_e} |\{r_{ie}=0\}|}{N_e}$, refers to the proportion of learners who answered the exercise e incorrectly on their first attempt, where N_e is the number of learners who answered exercise e, r_{ie} is the response (0 or 1). Referring to [4] which sets 10 difficulty levels based on learners' responses, we similarly map the error rate of exercises into 10 difficulty levels, i.e., $d(e) = \lfloor \phi(e) \times 10 \rfloor$. The difficulty matrix $\boldsymbol{D} \in \mathbb{R}^{10 \times d_r}$ encodes the 10 difficulty levels, d_r denotes its dimension. The difficulty embedding of exercise e^p_t is denoted as $\boldsymbol{d}^p_t \in \mathbb{R}^{d_r}$.

Interaction Embedding. We use a fully connected layer to deeply integrate the exercise embedding, difficulty embedding, and response embedding to obtain a basic interaction embedding. The interaction embedding $\boldsymbol{I}^p_t$ about solving exercise e^p_t is represented as follows:

$$\boldsymbol{I}^p_t = ReLU(\boldsymbol{W}_I[\boldsymbol{e}^p_t \oplus \boldsymbol{d}^p_t \oplus \boldsymbol{r}^p_t] + \boldsymbol{b}_I) \tag{2}$$

where $\boldsymbol{W}_I \in \mathbb{R}^{d_m \times (2 \times d_r + d_k)}$ and $\boldsymbol{b}_I \in \mathbb{R}^{d_m}$ are trainable parameters, and $\oplus$ represents the concatenation operation.

Behavior Embedding. Fine-grained learning behaviors can reflect learners' knowledge states. For example, when learners grasp the knowledge concepts well, they may not frequently submit answers and use the online hint function. We represent fine-grained learning behaviors (i.e., answer times, attempt counts, and hint counts) as embedding.

- *Time Embedding*. Following the approach proposed by [11], the embedding matrix $\boldsymbol{A_t}$ is used to represent discretized answer times which are measured in seconds. $\boldsymbol{A_t} \in \mathbb{R}^{d_{at} \times d_k}$, where d_{at} represents the number of discretized answer times and d_k represents the dimension. at^p_t is the answer time for solving exercise e^p_t and is represented as the vector $\boldsymbol{at}^p_t \in \mathbb{R}^{d_k}$.
- *Attempt Count Embedding*. The embedding matrix $\boldsymbol{A_c}$ represents the learners' attempt counts for the same exercise. $\boldsymbol{A_c} \in \mathbb{R}^{d_{ac} \times d_r}$, where d_{ac} represents the number of attempt counts, and d_r represents the dimension. ac^p_t represents the attempt counts for the learner solving exercise e^p_t and is represented as the vector $\boldsymbol{ac}^p_t \in \mathbb{R}^{d_r}$.
- *Hint Count Embedding*. Similarly, the embedding matrix $\boldsymbol{H_c}$ denotes the hint counts that the learner requests from the system for the same exercise. $\boldsymbol{H_c} \in \mathbb{R}^{d_{hc} \times d_r}$, where d_{hc} denotes the number of hints and d_r denotes the dimension. hc^p_t is hint counts request from system during learner solving exercise e^p_t and is denoted to the vector $\boldsymbol{hc}^p_t \in \mathbb{R}^{d_r}$.

To capture the effects of learning behaviors on the learning process, we also use a fully connected layer to integrate fine-grained behaviors (response time, hints count, attempts count) as behavior embedding. The behavior embedding $\boldsymbol{B}^p_t$ about answering exercise e^p_t is denoted as follows:

$$\boldsymbol{B}^p_t = ReLU(\boldsymbol{W}_b[\boldsymbol{at}^p_t \oplus \boldsymbol{hc}^p_t \oplus \boldsymbol{ac}^p_t] + \boldsymbol{b}_b) \tag{3}$$

where $\boldsymbol{B}^p_t$ represents the learners' behavior embedding when answering exercise e^p_t. $\boldsymbol{at}^p_t$, $\boldsymbol{hc}^p_t$ and $\boldsymbol{ac}^p_t$ represent the embeddings of the response time, hint counts, and attempt counts, respectively. $\boldsymbol{W}_b \in \mathbb{R}^{d_m \times (2d_r + d_k)}$ and $\boldsymbol{b}_b \in \mathbb{R}^{d_m}$ are trainable parameters.

Knowledge State Embedding. Following existing work [11,13], for each learner, we use the knowledge state embedding matrix $\boldsymbol{H}$ to store and update their knowledge states, $\boldsymbol{H} \in \mathbb{R}^{M \times d_m}$. M is the number of knowledge concepts. Each row of matrix $\boldsymbol{H}$ represents the learners' mastery level on the corresponding knowledge concept. In this article, we introduce $\boldsymbol{H^S}$ and $\boldsymbol{H^L}$ to represent learners' short-term and long-term knowledge states, respectively.

5.2. Modeling intra-session

For intra-session with observable interactions, fine-grained learning behaviors about interactions are used to model learners' short-term knowledge states. After finishing intra-session modeling in the pth session, the learners' short-term knowledge state $\boldsymbol{H}_p^{\boldsymbol{S}}$ is the input of inter-session modeling in offline time $O_{p,p+1}$.

5.2.1. Knowledge acquisition modeling

Learning acquisition refers to the differences in abilities and personal development exhibited by the same learner at two different times [41]. Existing work [13] suggests that learning behaviors have quite complex effects on learners' learning process, we model learners' knowledge acquisition by considering fine-grained learning behaviors.

Knowledge acquisition. While a learner answers exercise e_t^p, fine-grained learning behaviors may be generated, such as submitting answers quickly, constantly requesting hints from the system, or attempting frequently. These behaviors indirectly indicate learners' knowledge proficiency on exercise e_t^p. For example, if a learner frequently requests hints from the system or quickly submits answers, it may indicate a lack of engaging learning. The phenomena also reflect a lower knowledge proficiency. Therefore, it is necessary to consider the impact of guessing and engagement in modeling learners' knowledge acquisition. Based on the above analysis, we propose two gate mechanisms to model learners' fine-grained learning behaviors (i.e., answer time at_t^p, number of hints requested hc_t^p, and attempt counts ac_t^p) as indicators of guess and engage when answering exercise e_t^p.

$$G(\boldsymbol{e}_t^p) = sigmiod(\boldsymbol{W}_g[\boldsymbol{I}_t^p \oplus \boldsymbol{B}_t^p] + \boldsymbol{b}_g) \tag{4}$$

$$E(\boldsymbol{e}_t^p) = sigmoid(\boldsymbol{W}_e[\boldsymbol{I}_t^p \oplus \boldsymbol{B}_t^p] + \boldsymbol{b}_e) \tag{5}$$

where $G(\boldsymbol{e}_t^p)$ represents learners' guess and $E(\boldsymbol{e}_t^p)$ represents the learners' engagement level when answering exercise e_t^p. $\boldsymbol{I}_t^p$ and $\boldsymbol{B}_t^p$ are the interaction embedding and behavior embedding about e_t^p. $\boldsymbol{W}_g \in \mathbb{R}^{d_m \times 2d_m}$, $\boldsymbol{W}_e \in \mathbb{R}^{d_m \times 2d_m}$ are trainable weight matrices. $\boldsymbol{b}_g \in \mathbb{R}^{d_m}$ and $\boldsymbol{b}_e \in \mathbb{R}^{d_m}$ are trainable bias terms.

Considering learners' guess and engagement while answering the exercise, the learning acquisition after answering exercise e_t^p is calculated as follows:

$$\boldsymbol{LA}_t^p = relu(\boldsymbol{W}_a[\boldsymbol{I}_t^p \oplus \widehat{\boldsymbol{h}_{t-1}^p}] + \boldsymbol{b}_a) \tag{6}$$

$$\widehat{\boldsymbol{h}_{t-1}^p} = \left[(\boldsymbol{1} - G(\boldsymbol{e}_t^p)) \odot \boldsymbol{h}_{t-1}^p\right] \oplus \left[E(\boldsymbol{e}_t^p) \odot \boldsymbol{h}_{t-1}^p\right] \tag{7}$$

where $\boldsymbol{LA}_t^p$ represents the learning acquisition after the learners answer exercise e_t^p. $\widehat{\boldsymbol{h}_{t-1}^p}$ represents the usage of knowledge states during the learners' guess and engagement. $\odot$ represents the element-wise product. $\boldsymbol{W}_a$ and $\boldsymbol{b}_a$ are trainable parameters, where $\boldsymbol{W}_a \in \mathbb{R}^{d_m \times 3d_m}$, $\boldsymbol{b}_a \in \mathbb{R}^{d_m}$. $\boldsymbol{h}_{t-1}^p = \boldsymbol{q}_t^p \cdot \boldsymbol{H}_{p,t-1}^{\boldsymbol{S}}$ represents the knowledge proficiency related to solve exercise e_t^p. $\boldsymbol{q}_t^p$ is the knowledge concept vector about exercise e_t^p. $\boldsymbol{H}_{p,t-1}^{\boldsymbol{S}}$ is the learners' knowledge proficiency at step $t-1$ of session S^p.

Knowledge level increment quantification. The knowledge level increment on all knowledge is quantified based on the learning acquisition $\boldsymbol{LA}_t^p$ after learners' answering exercise e_t^p. We multiply the knowledge concept vector $\boldsymbol{q}_t^p$ of exercise e_t^p with the learning acquisition $\boldsymbol{LA}_t^p$ to quantify the knowledge level increment on all knowledge as follows:

$$\boldsymbol{LI}_t^p = \boldsymbol{q}_t^p \cdot \boldsymbol{LA}_t^p \tag{8}$$

5.2.2. Knowledge forgetting modeling

While learners acquire new knowledge, they also forget previously learned knowledge. Prior work [13] suggests that knowledge forgetting is influenced by time and learners' behaviors. If learners frequently guess answers while answering exercises, they will not recall or apply their knowledge. Consequently, their knowledge proficiency will decrease with time. Therefore, integrating learners' previous knowledge states and the guess and engagement levels while answering the current exercise e_t^p, knowledge forgetting is calculated as follows:

$$\boldsymbol{LF}_t^p = sigmoid(\boldsymbol{W}_f[\boldsymbol{H}_{p,t-1}^{\boldsymbol{S}} \oplus \boldsymbol{I}_t^p \oplus \widehat{\boldsymbol{h}_{t-1}^p}] + \boldsymbol{b}_f) \tag{9}$$

where $\boldsymbol{LF}_t^p$ represents the forgetting on the learners' previous knowledge state $\boldsymbol{H}_{p,t-1}^{\boldsymbol{S}}$ after answering exercise e_t^p. $\boldsymbol{W}_f$ and $\boldsymbol{b}_f$ are trainable parameters, where $\boldsymbol{W}_f \in \mathbb{R}^{d_m \times 4d_m}$, $\boldsymbol{b}_f \in \mathbb{R}^{d_m}$.

5.2.3. Knowledge update modeling

When each interaction within the pth session is finished, learners' knowledge state is updated accordingly. Based on learners' knowledge level increment and the knowledge forgetting after answering exercise e_t^p, the current knowledge state $\boldsymbol{H}_{p,t}^{\boldsymbol{S}}$ is updated as follows:

$$\boldsymbol{H}_{p,t}^{\boldsymbol{S}} = \boldsymbol{LI}_t^p + (\boldsymbol{1} - \boldsymbol{LF}_t^p) \odot \boldsymbol{H}_{p,t-1}^{\boldsymbol{S}} \tag{10}$$

After all interactions within the pth session are finished, learners' knowledge state at the last step of the pth session $\boldsymbol{H}_{p,last}^{\boldsymbol{S}}$ represent their short-term knowledge state in the pth session. For convenience, we use $\boldsymbol{H}_p^{\boldsymbol{S}}$ to replace $\boldsymbol{H}_{p,last}^{\boldsymbol{S}}$.

5.3. Modeling inter-session

In offline time, knowledge retention and decay are important factors affecting knowledge state shifts. For inter-session modeling, we model the knowledge retentions and decays to capture the knowledge state shifts between sessions from the knowledge concept level. Meanwhile, the frequency of knowledge practice is used to control the retention rate of knowledge and filter poorly mastered knowledge.

5.3.1. Short-to-long knowledge retention

According to education psychology theories, part of learners' short-term learning memory gradually consolidates into their long-term memory, where the retention time is very long [42]. In inter-sessions without observable interactions, we model learners' short-to-long knowledge state retention to predict learners' performance in the next session. If learners have a high retention of knowledge state, the knowledge state shifts between sessions will be less. Therefore, they will have a high probability of correctly answering the relevant exercises in the next session.

In the offline time $O_{p,p+1}$ between the pth and p+1th sessions, a GRU cell is used to consolidate the short-term knowledge state $\boldsymbol{H}_p^{\boldsymbol{S}}$ into long-term knowledge $\boldsymbol{H}_p^{\boldsymbol{L}}$. Educational psychology theories also indicate that repeated retrieval can strengthen memory retention [28,43,44]. Therefore, the frequency of knowledge practice in the pth session is used to control the retention rate of short-term knowledge state in consolidating. The process is as follows:

$$\widetilde{\boldsymbol{H}_p^{\boldsymbol{S}}} = sigmoid(\boldsymbol{W}_c \boldsymbol{F}_q + \boldsymbol{b}_c) \odot \boldsymbol{H}_p^{\boldsymbol{S}} \tag{11}$$

$$\boldsymbol{R} = sigmoid(\boldsymbol{W}_r[\widetilde{\boldsymbol{H}_p^{\boldsymbol{S}}} \oplus \boldsymbol{H}_{p-1}^{\boldsymbol{L}}] + \boldsymbol{b}_r) \tag{12}$$

$$\boldsymbol{Z} = sigmoid(\boldsymbol{W}_z[\widetilde{\boldsymbol{H}_p^{\boldsymbol{S}}} \oplus \boldsymbol{H}_{p-1}^{\boldsymbol{L}}] + \boldsymbol{b}_z) \tag{13}$$

$$\widetilde{\boldsymbol{H}_p^{\boldsymbol{L}}} = \tanh(\boldsymbol{W}_h[\widetilde{\boldsymbol{H}_p^{\boldsymbol{S}}} \oplus (\boldsymbol{R} \odot \boldsymbol{H}_{p-1}^{\boldsymbol{L}})] + \boldsymbol{b}_h) \tag{14}$$

$$\boldsymbol{H}_p^{\boldsymbol{L}} = \boldsymbol{Z} \odot \boldsymbol{H}_{p-1}^{\boldsymbol{L}} + (\boldsymbol{1} - \boldsymbol{Z}) \odot \widetilde{\boldsymbol{H}_p^{\boldsymbol{L}}} \tag{15}$$

where $\boldsymbol{H}_p^{\boldsymbol{L}}$ is the long-term knowledge state during offline time $O_{p,p+1}$. $\boldsymbol{H}_p^{\boldsymbol{S}}$ and $\widetilde{\boldsymbol{H}_p^{\boldsymbol{S}}}$ are the short-term knowledge states acquired from the pth session. $\boldsymbol{F}_q$ is the vector of knowledge practice frequency in the pth session. $\boldsymbol{H}_{p-1}^{\boldsymbol{L}}$ is learners' long-term knowledge states during offline time $O_{p-1,p}$. $\widetilde{\boldsymbol{H}_p^{\boldsymbol{L}}}$ represents the candidate long-term knowledge state. $\boldsymbol{R}$ and $\boldsymbol{Z}$ denote the forget gate and update gate, respectively. $\boldsymbol{W}_c \in \mathbb{R}^{M \times M}$, $\boldsymbol{W}_r \in \mathbb{R}^{d_m \times 2d_m}$, $\boldsymbol{W}_z \in \mathbb{R}^{d_m \times 2d_m}$, $\boldsymbol{W}_h \in \mathbb{R}^{d_m \times 2d_m}$, $\boldsymbol{b}_c \in \mathbb{R}^M$, $\boldsymbol{b}_r \in \mathbb{R}^{d_m}$, $\boldsymbol{b}_z \in \mathbb{R}^{d_m}$ and $\boldsymbol{b}_h \in \mathbb{R}^{d_m}$ are trainable parameters.

5.3.2. *Short-term knowledge decay*

Learners' short-term learning memory is an intermediate state between immediate and long-term memory, which decays over time [45]. In inter-sessions, we also model the short-term knowledge decay $\boldsymbol{H}_p^{\boldsymbol{S}}$ in offline time $O_{p,p+1}$ to predict learners' performance in the next session. The frequency of knowledge practice has an effect on short-term knowledge decay [28,43]. The higher the frequency of knowledge practice, the less knowledge decay in the short-term knowledge state.

In offline time $O_{p,p+1}$, the exponential decay function is used to simulate the decay of learners' short-term knowledge state $\boldsymbol{H}_p^{\boldsymbol{S}}$, which has been widely used in knowledge tracing and has proven effective for modeling memory decay. Additionally, learners' knowledge practice frequency within the pth session controls the decay rate of the short-term knowledge state over time. The calculation is as follows:

$$\overline{\boldsymbol{H}_p^{\boldsymbol{S}}} = exp(-\left|O_{p,p+1}\right|(\boldsymbol{W}_s \boldsymbol{F}_q)) \odot \boldsymbol{H}_p^{\boldsymbol{S}} \tag{16}$$

where $\overline{\boldsymbol{H}_p^{\boldsymbol{S}}}$ denotes the short-term knowledge state after decay over the offline time $\left|O_{p,p+1}\right|$. $\boldsymbol{W}_s \in \mathbb{R}^{M \times M}$ is trainable parameters. After calculating the knowledge retention and decay in offline time $O_{p,p+1}$, the fused short- and long-term knowledge state is the initial knowledge state $\boldsymbol{H}_{p+1,0}^{\boldsymbol{S}}$ of the next $p+1$th session. The calculation is as follows, and $\alpha \in [0,1]$ is a hyper-parameter.

$$\boldsymbol{H}_{p+1,0}^{\boldsymbol{S}} = \alpha \boldsymbol{H}_p^{\boldsymbol{L}} + (1-\alpha)\overline{\boldsymbol{H}_p^{\boldsymbol{S}}} \tag{17}$$

5.4. *Performance prediction*

In this part, we demonstrate how to use the learners' knowledge state $\boldsymbol{H}_{p,t}^{\boldsymbol{S}}$ to predict their performance on the next exercise e_{t+1}^p. In online learning, when learners solve exercise e_{t+1}^p with the difficulty level d_{t+1}^p, we infer the learners' performance on exercise e_{t+1}^p as follows:

$$y_{t+1}^p = sigmoid(\boldsymbol{W}_o[\boldsymbol{e}_{t+1}^p \oplus \boldsymbol{d}_{t+1}^p \oplus \boldsymbol{h}_t^p] + \boldsymbol{b}_o) \tag{18}$$

where y_{t+1}^p is the predicted probability of learners' correctly answering exercise e_{t+1}^p, which ranges from 0 to 1. $\boldsymbol{h}_t^p = \boldsymbol{q}_{t+1}^p \cdot \boldsymbol{H}_{p,t}^{\boldsymbol{S}}$ represents learners' knowledge levels related to exercise e_{t+1}^p. $\boldsymbol{W}_o \in \mathbb{R}^{d_m \times (d_k + d_r + d_m)}$ and $\boldsymbol{b}_o \in \mathbb{R}^{d_m}$ are trainable parameters. To train all the parameters in ELPKT, we exploit the cross-entropy loss between the predicted response y and the true response r as the objective function, which will be minimized in the training process:

$$L = -\sum_{p=2}^{P}\sum_{t=1}^{T}(r_t^p \log y_t^p + (1-r_t^p)\log(1-y_t^p)) \tag{19}$$

6. Experiment

In this section, we conduct extensive experiments to evaluate the proposed ELPKT model, aiming to answer the following essential research questions:

> **RQ1**: How does our proposed ELPKT model perform against the state-of-the-art baseline models?
> **RQ2**: How do the components (i.e., forgetting, inter-session modeling, the components of inter-session modeling, and the fusion of short-term and long-term knowledge) within ELPKT and interval thresholds for splitting sessions affect the model performance?
> **RQ3**: Can our proposed ELPKT model capture the knowledge state shifts between sessions and how does it perform in fine-grained behavior modeling as well as providing interpretability for learners' performance prediction?

To answer these questions, we first provide the details of the experimental setup, including the data processing, training details, evaluation, and baseline methods. Afterward, we present the KT models' performance to answer question **RQ1**. Then, we conduct an ablation study and parameter sensitivity analysis to answer question **RQ2**. Furthermore, we adopt a case study to answer questions **RQ3**.

6.1. *Experimental setup*

6.1.1. *Training details*

To facilitate training, we formulate learners' complete learning processes with P sessions and $P-1$ offline time, where each session contains T interactions. To approximate learners' authentic session-based learning experiences, we set P and T to the third quartile of the session counts and the session length in datasets to represent learners' learning processes. Specifically, as shown in Table 2, for ASSIST2012, P and T are set to 14 and 14; For ASSIST2017, P and T are 10 and 86; For MOOC746997, P and T are 10 and 14; For MOOC770738, P and T are 19 and 5. For the sessions with lengths greater than T, we divided them into multiple sessions with fixed-length T. For learners with more sessions than P, we sliced their sessions based on the fixed count P. For sessions whose length is smaller than T or learners' session counts less than P, we padded them with zero vectors.

We initialized all parameters using a uniform distribution [46]. All parameters were learned during the training process. We set the mini-batch size to 16 in our experiments. The parameters d_k, d_m, and d_r were set to 128, 128, and 50, respectively. The hyper-parameter α for short- and long-term knowledge fusion was set to 0.3. The initial learning rate was set to 0.001 and decayed after each epoch. The optimizer is Adam [47]. We apply the early stopping strategy to cut off the training when the AUC on the validation sets does not grow in 5 consecutive epochs. All baselines were carefully tuned to achieve optimal performance to ensure fairness. All experiments were conducted on a Linux server with an RTX 4090 GPU. We conducted five independent runs and reported the average results for all models.

6.1.2. *Evaluation*

To comprehensively evaluate the performance of all models, we conducted experiments on four benchmark datasets. For all models, we performed 5-fold cross-validation on all datasets. For each fold, 20% of the learners were used as the test set, and the remaining 80% were divided into 80% for training and 20% for validation. The model with the best performance on the validation set was used to evaluate the test set. We report the average results of five runs on the test set. **Area under curve (AUC)**, **accuracy (ACC)** and **Root mean squared error (RMSE)** are the evaluation metrics that are commonly used in the KT task. Specifically, AUC and ACC are adopted to measure the model's effectiveness from the classification perspective. RMSE is used to quantify the distance between the predicted and actual performance.

6.2. *Experimental results and discussion*

6.2.1. *Baselines*

To evaluate the effectiveness of the ELPKT model, we compare it with nine representative KT models. For better presentation, we summarize the properties of the baselines and the ELPKT model in Table 3 and divide the baselines into three categories as follows:

(1) *The representative KT works that do not consider temporal effects on learners' knowledge state.*

- **DKT** [18] uses Recurrent Neural Networks (RNNs) to model learners' knowledge states at each time step.
- **DKT+** [19] addresses the reconstruction error and the waveform transition in the DKT model.
- **DKVMN** [20] is a memory-augmented KT model. It utilizes the relationships of latent concepts to output the learners' knowledge mastery levels.

(2) *The works modeling temporal effects in a unified way.*

- **LPKT** [11] considers the answering time and time interval of learners' interactions to calculate learning gain and forgetting.

Table 3
Properties of all baselines and our ELPKT model.

Method	Session-aware	Time interval	Fine-grained behavior	Knowledge state representation
DKT	–	–	–	A single vector
DKT+	–	–	–	A single vector
DKVMN	–	–	–	Knowledge state matrix
AKT	–	✓	–	A single vector
HawkesKT	–	✓	–	Intensity value
LPKT	–	✓	Answer time	Knowledge state matrix
LBKT	–	✓	Answer time, hint count, attempt count	Knowledge state matrix
QKT	✓	–	–	A single vector
HiTSKT	✓	✓	–	A single vector
ELPKT	✓	✓	Answer time, hint count, attempt count	Knowledge state matrix

Table 4
The average results comparing ELPKT with the representative KT models over all datasets.

Methods	ASSIST2012			ASSIST2017			MOOC746997			MOOC770738		
	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE
DKT	0.7054	0.7234	0.4359	0.698	0.6855	0.4483	0.7896	0.7614	0.4285	0.7826	0.867	0.3645
DKT+	0.7135	0.7289	0.4337	0.682	0.6943	0.4392	0.7901	0.7689	0.4157	0.8013	0.8491	0.3325
DKVMN	0.7204	0.6933	0.4280	0.6853	0.7062	0.4263	0.7573	0.7634	0.431	0.8233	0.8476	0.3559
Improve	8.8%	4.1%	5.2%	18.7%	7.9%	5.7%	3%	0.8%	5.2%	1.5%	2.7%	6.6%
AKT	0.7641	0.7505	0.4123	0.7624	0.7135	0.4326	0.8053	0.7679	0.3999	0.8284	**0.8741**	0.3175
HawkesKT	0.7571	0.7470	0.4153	0.7052	0.6868	0.4531	0.7934	0.7616	0.4061	0.8257	0.8694	0.3154
LPKT	0.7734	0.7549	0.4951	0.7962	0.7371	0.4828	0.8055	0.769	0.4806	0.829	0.8709	0.3569
LBKT	0.7748	0.7561	0.4097	0.7958	0.7387	0.4192	0.8103	0.7736	0.3959	0.8312	0.8712	**0.3087**
Improve	1.2%	0.4%	0.9%	4%	3.2%	4.2%	0.4%	0.2%	0.5%	0.5%	−0.3%	−0.6%
QKT	0.7127	0.7276	0.4280	0.7105	0.6887	0.4509	0.8053	0.7665	0.4	0.7712	0.8540	0.3353
HiTSKT	0.7546	0.7442	0.5058	0.7817	0.7224	0.5269	0.8059	0.7677	0.4819	0.8319	0.8637	0.3687
HiTSKT†	0.7566	0.7467	0.5033	0.7901	0.7294	0.5202	0.8087	0.7714	0.4675	0.8324	0.8661	0.3614
ELPKT	**0.7844**	**0.7591**	**0.4059**	**0.8285**	**0.7620**	**0.4018**	**0.8138**	**0.7750**	**0.394**	**0.8357**	0.8719	0.3106
Improve	3.9%	2%	5.2%	6%	5.5%	10.9%	1%	1%	1.5%	0.5%	0.9%	7.4%

The best result in each column is in bold, and the best result in each category of models is underlined. HiTSKT and HiTSKT† denote the experimental results based on sessions split by 10 h and 30 min, respectively. *Improve* represents the improvement of ELPKT over the best baselines of each category.

- **AKT** [10] summarizes learners' historical performances using a monotonic attention mechanism.
- **HawkesKT** [12] models the temporal cross-effects on skill mastery by point processes.
- **LBKT** [13] explores the learners' behavior effects on the learning gain and forgetting.

(3) *The session-aware knowledge tracing works*.

- **QKT** [16] splits quizzes by quiz ID and models the intra- and the inter-quiz to trace knowledge states. Considering the quizzes split by quiz IDs may have inconsistent data distribution from the sessions split by time interval, we run it in the same experimental setup as ours.
- **HiTSKT** [17] includes an interaction-level encoder and a session-level encoder. It splits learners' sequences into sessions when the time interval between adjacent interactions is greater than 10 h.
- **HiTSKT**† [17] run in the sessions that are split by our default interval threshold (i.e., 30 min), with the same experimental setup as ours.

6.2.2. Performance prediction

To answer the research question **RQ1**, this experiment compares the performance of the proposed ELPKT model with nine representative KT baseline models on four datasets. Table 4 reports the average experiment results of five runs. From Table 4, we have the following findings:

ELPKT outperforms the representative baselines that do not consider temporal effects. Compared with the best baseline (i.e., the best among DKT, DKT+, and DKVMN on AUC, ACC, and RMSE, respectively) without considering temporal effects, ELPKT achieved better learners' performance prediction in four datasets. The results show that ELPKT, considering the temporal effect, can effectively track learners' knowledge states.

ELPKT outperforms the baselines that model temporal effects in a unified way. Compared with these baselines, ELPKT performed better in the ASSIST2012, ASSIST2017, and MOOC746997 datasets on three metrics. The results validate that modeling large and small time intervals in different ways benefits in predicting learners' performance. For MOOC770738, our ELPKT performs slightly worse than AKT and LBKT on ACC and RMSE. This may be because the MOOC datasets lack detailed behavior features, e.g., hint counts and repeated attempt counts, which affect model performance. Besides, in MOOC770738, there are fewer interactions in most learners' sessions (i.e., the number of interactions in most sessions is 5, as listed in Table 2), which may hinder the ELPKT model from roundly capturing knowledge state within sessions.

ELPKT outperforms the representative session-aware KT baseline models. Compared with HiTSKT and QKT, ELPKT improved significantly in the four datasets. It validates that considering fine-grained behavior and the knowledge state shifts between sessions at the knowledge concept level benefits tracing learners' knowledge state. Moreover, the performance of the HiTSKT† running in sessions split by the small interval threshold (i.e., 30 min) was better than HiTSKT running in sessions split by 10 h. It demonstrates that sessions split by too large thresholds may hinder the prediction ability of session-aware KT models.

Other important features contribute to KT modeling. In Table 4, we observe that the baselines considering temporal effects (e.g., AKT, HawkesKT, LPKT, HiTSKT, LBKT) and fine-grained behavior effect (e.g., LBKT) present better performance than those that do not consider temporal effects. Moreover, LBKT, which considers both temporal and behavioral effects, performs best on almost all baselines. This proves

Table 5
Setting differences between ELPKT and its variant methods.

Line	Methods	intra-session	inter-session			
		forgetting	retention	frequency on retention	decay	frequency on decay
M0	ELPKT	✓	✓	✓	✓	✓
M1	**w/o forget**	–	✓	✓	✓	✓
M2	**w/o offline**	✓	–	–	–	–
M3	**w/o retention**	✓	–	–	✓	✓
M4	**w/o Feq_retention**	✓	✓	–	✓	✓
M5	**w/o decay**	✓	✓	✓	–	–
M6	**w/o Feq_decay**	✓	✓	✓	✓	–

Table 6
PerformancecComparison between ELPKT and its variant methods (under %).

Line	Methods	ASSIST2012			ASSIST2017			MOOC746997			MOOC770738		
		AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE
M0	ELPKT	**78.44**	**79.51**	**40.59**	**82.85**	**76.2**	**40.18**	**81.38**	**77.50**	**39.4**	**83.57**	**87.19**	**31.06**
M1	**w/o forget**	77.1	75.0	41.18	80.46	74.53	41.49	81.02	77.39	39.47	82.2	87.05	31.48
M2	**w/o offline**	76.91	74.85	41.73	79.83	74.04	41.68	80.35	77.13	39.62	81.5	86.59	31.57
M3	**w/o retention**	78.25	75.78	40.69	79.97	74.29	41.47	81.23	77.21	39.53	83.16	86.86	31.12
M4	**w/o Feq_retention**	78.31	75.85	40.6	81.39	75.19	40.99	81.36	77.34	39.49	83.48	86.92	31.1
M5	**w/o decay**	78.38	75.87	40.63	82.37	75.79	40.44	81.29	77.39	39.43	83.49	87.11	31.15
M6	**w/o Feq_decay**	78.4	75.88	40.61	82.48	75.93	40.39	81.33	77.45	39.41	83.52	87.15	31.09
B0	**w/o offline**	69.96	71.98	43.39	68.77	66.96	46.12	79.56	75.46	40.99	75.50	85.29	33.98
B1	**w/o offline**	74.66	73.69	51.30	76.97	71.29	53.58	79.91	76.39	47.01	81.21	86.14	36.71

$B0$ and $B1$ are the session-aware KT baselines (i.e., QKT [16] and HiTSKT [17]) that remove inter-session modeling.

Table 7
The overall and partial performance comparisons of ELPKT with representative session-aware KT baselines.

Methods	ASSIST2012			ASSIST2017			MOOC746997			MOOC770738		
	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE
ELPKT	0.7844	0.7591	0.4059	0.8285	0.7620	0.4018	0.8138	0.7750	0.394	0.8357	0.8719	0.3106
QKT	0.7127	0.7276	0.4280	0.7105	0.6887	0.4509	0.8053	0.7665	0.40	0.7712	0.8540	0.3353
HiTSKT	0.7566	0.7467	0.5033	0.7901	0.7294	0.5202	0.8087	0.7714	0.4675	0.8324	0.8661	0.3614
$Improve1$	10.06%	4.33%	5.16%	16.60%	10.64%	10.89%	1.06%	1.11%	1.50%	8.36%	2.10%	7.37%
$Improve2$	3.67%	1.66%	19.35%	4.86%	4.47%	22.76%	0.63%	0.47%	15.72%	0.40%	0.67%	14.06%
M2	0.7691	0.7485	0.4173	0.7983	0.7404	0.4168	0.8035	0.7713	0.3962	0.815	0.8659	0.3157
B0	0.6996	0.7198	0.4339	0.6877	0.6696	0.4612	0.7956	0.7546	0.4099	0.7550	0.8529	0.3398
B1	0.7466	0.7369	0.5130	0.7697	0.7129	0.5358	0.7991	0.7639	0.4701	0.8121	0.8614	0.3671
$Improve3$	9.93%	3.99%	3.83%	16.08%	10.57%	9.63%	0.99%	2.21%	3.34%	7.95%	1.52%	7.09%
$Improve4$	3.01%	1.57%	18.65%	3.72%	3.86%	22.21%	0.55%	0.97%	15.72%	0.36%	0.52%	14.0%

$M2$, $B0$, and $B1$ are the ELPKT, QKT [16] and HiTSKT [17] that remove inter-session modeling respectively. $Improve1$ and $Improve2$ denote the improvement of the proposed ELPKT over QKT and HiTSKT. $Improve3$ and $Improve4$ denote the improvement of $M2$ over $B0$ and $B1$.

that temporal information and fine-grained behavior are crucial for knowledge tracing.

All models performed better on the MOOC datasets than on ASSISTments datasets. It may be because most learners' online learning is regular and periodic in the MOOC datasets (i.e., most learners' offline time in MOOC datasets is much shorter than that in ASSISTments datasets, as seen in Fig. 3), which benefits to modeling their learning progress and knowledge state evolution.

6.2.3. Ablation studies

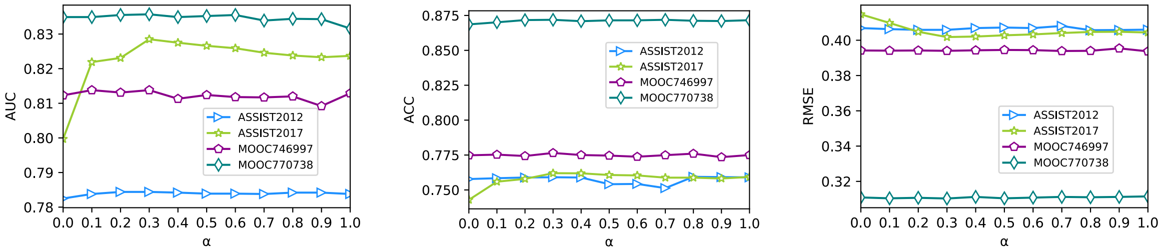
To answer the research question **RQ2**, we design 6 variants of the ELPKT model and the variants of QKT [16] and HiTSKT [17] for ablation studies as follows:

- **w/o forget** removes the component that models knowledge forgetting in intra-sessions.
- **w/o offline** removes the component that models inter-sessions. It is similar to most KT works that do not split learners' sequences into sessions.
- **w/o retention** removes the short-to-long knowledge retention in inter-session modeling.
- **w/o Feq_retention** neglects the effect of knowledge practice frequency on knowledge retention.
- **w/o decay** removes the decay of the short-term knowledge state over time in inter-session modeling.
- **w/o Feq_decay** neglects the effect of knowledge practice frequency on short-term knowledge decay.

Notably, both HiTSKT [17] and our ELPKT split sessions by the time interval, while QKT [16] splits quizzes (sessions) by quiz ID, which may have inconsistent data distribution with sessions divided by the time interval. To ensure fair comparisons, we run the variants of ELPKT, QKT, and HiTSKT at the same experiment settings, i.e., the variants of three models are run in the sessions split by the default threshold (30 min). For the input length (i.e., session counts and interaction counts within sessions) of the variants of the three models, we use the same training details as reported in Section 6.1.1.

Table 5 summarizes the difference in the settings of these variant methods, and Table 6 shows the performance comparison between them and ELPKT. Table 7 reports the performance of the session-aware KT models. From Table 6, we have the following findings:

Considering knowledge forgetting is essential for intra-session modeling. Compared with $M0$, $M1$ that removes knowledge forgetting in intra-session modeling significantly degrades performance. The result shows that knowledge forgetting occurs within sessions when knowledge is not applied.



(a) AUC (b) ACC (c) RMSE

Fig. 6. ELPKT performances in the different fusion ratios of short-term and long-term knowledge α.

Considering offline inter-session benefits enhancing learning process modeling. Compared with $M0$, $M2$ that removes inter-session modeling exhibits a significant decline in performance. The result demonstrates that considering the effects of large intervals on the knowledge state helps capture the knowledge state shifts during the learning process.

Knowledge retention and decay are important for inter-session modeling. Compared with $M0$, $M3$ that removes knowledge retention in inter-session modeling shows a decline. It suggests that learners' knowledge will be consolidated after a learning session. In addition, compared with $M0$, $M5$ that neglects knowledge decay in inter-session modeling also shows a decrease. It validates that learners' short-term knowledge will decay over time. Compared with $M3$, $M5$ drops a bit less. The reason may be that knowledge retention modeling implicitly involves knowledge filtering, where only well-mastered knowledge is retained for long-term knowledge. It leads to the effect of short-term knowledge decay being smaller than knowledge retention.

The frequency of knowledge practice affects both knowledge retention and decay. Compared with $M0$, $M4$ and $M6$, which neglect the effect of knowledge practice frequency on knowledge retention and knowledge decay, respectively, show a slight decrease. The results suggest that the frequency of knowledge practice positively contributes to knowledge tracing.

Comparison of the intra-session modeling in session-aware KT models. Compared with $B0$ and $B1$ (i.e., QKT and HiTSKT that remove inter-session modeling) in Table 6, $M2$ that removing the intra-session modeling in ELPKT outperforms both. It illustrates that considering learners' fine-grained learning behavior in intra-session modeling is more beneficial to accurately capture learners' knowledge states than only considering the relationship between interactions.

Comparison of the inter-session modeling in session-aware KT models. Given the inter-session modeling is built on the intra-session modeling, we evaluate the performance of inter-session modeling in three methods (i.e., ELPKT, QKT, and HiTSKT) by calculating the overall and partial performance improvement, as shown in Table 7. From Table 7, we find that: (1) Our ELPKT outperforms the representative session-aware KT baseline models, i.e., QKT and HiTSKT; (2) When only modeling intra-session in these three models, ELPKT also demonstrates its superior performance. (3) The overall improvement of ELPKT over QKT, and HiTSKT (i.e., $Improve1$ and $Improve2$) is higher than that of only modeling intra-sessions (i.e., $Improve3$ and $Improve4$), in the datasets except MOOC746997. On MOOC746997, while $Improve1$ and $Improve2$ show slightly lower ACC compared to $Improve3$ and $Improve4$, they outperform $Improve3$ and $Improve4$ in AUC which offers a more comprehensive evaluation of the model performance than ACC. As such, the results validate the effectiveness of inter-session modeling and capturing the knowledge state shifts between sessions in ELPKT.

6.2.4. Parameter sensitivity analysis

We conduct parameter sensitivity analysis further to answer the research question **RQ2**. It contains the study of the fusion of short- and long-term knowledge states and the interval threshold setting for splitting sessions.

The fusion of short-term and long-term knowledge states. We run ELPKT with the parameter $\alpha \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$, which is the fusion ratio of short-term and long-term knowledge as shown in Eq. (17), to determine the most favorable α for optimizing model performance. Fig. 6 shows the experimental results on four datasets. In Fig. 6, we find that our model performance shows a decline when we only consider the decayed short-term knowledge as the initial knowledge state of the next session, i.e., α is set to 0. It indicates that long-term knowledge can be propagated to the next session. Similarly, our model performance is not optimal when we only consider the long-term knowledge, i.e., α is set to 1. We conclude that both the learners' long-term and short-term knowledge states impact their learning performance.

Overall, our model performs best as α grows to 0.3. After that, its performance is relatively stable and not particularly sensitive to the fusion of long- and short-term knowledge. Therefore, α is set to 0.3 to fuse the learners' knowledge states before it is propagated to the next session.

The interval threshold setting for splitting sessions. Given the sessions split by too large thresholds (e.g., 10 h) do not align with learners' short online sessions, as analyzed in Section 4.3, we run ELPKT in the sessions split by small thresholds (i.e., $\theta \in \{20, 30, 40, 50\}$) to illustrate the effects of thresholds on the model performance.

The experimental results in Table 8 indicate that ELPKT performs better in sessions split by $\theta \in \{20, 30\}$ than in those split by $\theta \in \{40, 50\}$. ELPKT performs best in sessions split by $\theta = 30$ min in ASSIST2017 and MOOC770738, while there are some minor fluctuations in ASSIST2012 and MOOC746997.

6.2.5. A case study: The knowledge state evolution

Finally, to answer the research questions **RQ3**, we visualize a learner's knowledge state evolution. We assess the learner's knowledge level following the approach of [13]. At time step t of the pth session, the learner's level $y^m_{p,t}$ on the knowledge concept c_m is calculated as follows:

$$y^m_{p,t} = sigmoid(\boldsymbol{W}_o[\boldsymbol{h}^m_{p,t} \oplus \boldsymbol{0}] + \boldsymbol{b}_o) \tag{20}$$

where $\boldsymbol{h}^m_{p,t}$ is the knowledge state related to KC c_m. $\boldsymbol{0}$ represents the zero vector, whose dimension is equal to that of the exercise and difficulty level in Eq. (18). $\boldsymbol{W}_o$ and $\boldsymbol{b}_o$ are trained parameters in Eq. (18).

To validate the proposed ELPKT in capturing knowledge state shifts between sessions, we calculate the average difference in knowledge state evolution as follows:

$$dif_{intra} = \frac{\sum_{t=1}^{T-1} \left| y^m_{p,t+1} - y^m_{p,t} \right|}{T-1} \tag{21}$$

$$dif_{inter} = \frac{\sum_{t=1}^{T} \left| y^m_{p+1,t} - y^m_{p,T} \right|}{T} \tag{22}$$

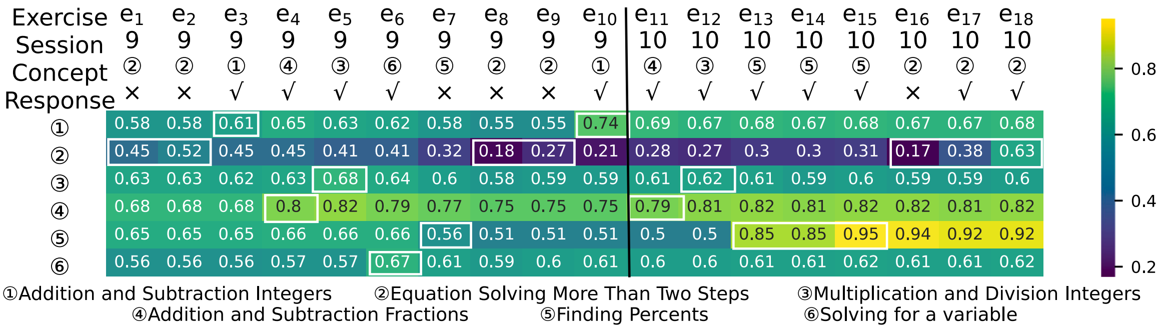
where dif_{intra} and dif_{inter} denote the average difference in knowledge state evolution on the same KCs within and between sessions, respectively. T represents all time steps in sessions. $y^m_{p,t}$ represents the learner's knowledge level on KC c_m at time step t of the pth session.

Table 8
The average results of ELPKT running in the sessions split by different interval threshold θ.

Methods	ASSIST2012			ASSIST2017			MOOC746997			MOOC770738		
	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE	AUC	ACC	RMSE
θ=20	0.7852	0.7603	0.4050	0.8115	0.7481	0.4107	0.8165	0.7833	0.3903	0.8316	0.8623	0.3218
θ=30	0.7844	0.7591	0.4059	0.8285	0.7620	0.4018	0.8138	0.7750	0.394	0.8357	0.8719	0.3106
θ=40	0.7821	0.7535	0.4068	0.8211	0.7565	0.4054	0.8114	0.7741	0.3951	0.8340	0.8681	0.3155
θ=50	0.7801	0.7502	0.4213	0.8037	0.7434	0.4141	0.8087	0.7692	0.3967	0.8326	0.8565	0.3254

Table 9
The details of the learner's interactions in the 9th and 10th sessions.

Exercise KC	e_1 ②	e_2 ②	e_3 ①	e_4 ④	e_5 ③	e_6 ⑥	e_7 ⑤	e_8 ②	e_9 ②	e_{10} ①	e_{11} ④	e_{12} ③	e_{13} ⑤	e_{14} ⑤	e_{15} ⑤	e_{16} ②	e_{17} ②	e_{18} ②
Difficulty	5	8	4	2	1	4	5	4	4	2	3	1	6	2	5	5	3	3
Answer time (s)	136	249	10	28	4	313	142	20	28	4	15	5	21	6	19	35	30	25
Hint	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0
Attempt	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1



Exercise	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_{15}	e_{16}	e_{17}	e_{18}
Session	9	9	9	9	9	9	9	9	9	9	10	10	10	10	10	10	10	10
Concept	②	②	①	④	③	⑥	⑤	②	②	①	④	③	⑤	⑤	⑤	②	②	②
Response	×	×	√	√	√	√	×	×	×	√	√	√	√	√	√	×	√	√
①	0.58	0.58	0.61	0.65	0.63	0.62	0.58	0.55	0.55	0.74	0.69	0.67	0.68	0.67	0.68	0.67	0.67	0.68
②	0.45	0.52	0.45	0.45	0.41	0.41	0.32	0.18	0.27	0.21	0.28	0.27	0.3	0.3	0.31	0.17	0.38	0.63
③	0.63	0.63	0.62	0.63	0.68	0.64	0.6	0.58	0.59	0.59	0.61	0.62	0.61	0.59	0.6	0.59	0.59	0.6
④	0.68	0.68	0.68	0.8	0.82	0.79	0.77	0.75	0.75	0.75	0.79	0.81	0.82	0.81	0.82	0.82	0.81	0.82
⑤	0.65	0.65	0.65	0.66	0.66	0.66	0.56	0.51	0.51	0.51	0.5	0.5	0.85	0.85	0.95	0.94	0.92	0.92
⑥	0.56	0.56	0.56	0.57	0.57	0.67	0.61	0.59	0.6	0.61	0.6	0.6	0.61	0.61	0.62	0.61	0.61	0.62

0.8
0.6
0.4
0.2
①Addition and Subtraction Integers
②Equation Solving More Than Two Steps
③Multiplication and Division Integers
④Addition and Subtraction Fractions
⑤Finding Percents
⑥Solving for a variable

Fig. 7. A learner's knowledge state evolution in the 9th and 10th sessions.

We visualize an example from the ASSIST2012 dataset in Fig. 7, which displays the learner's knowledge state in the 9th and 10th sessions traced by ELPKT. We also show the details of the learner's interactions in the 9th and 10th sessions in Table 9. From Fig. 7 and Table 9, we have the following findings:

ELPKT can capture knowledge state shifts between sessions. Knowledge state shifts between sessions denote that learners' performances on the same knowledge concepts may be different in adjacent sessions. In Fig. 7, there is a significant shift in the learner's performances on KCs "②" and "⑤" between the 9th and 10th sessions. To validate whether our ELPKT can capture knowledge state shifts between sessions, we calculate the average differences in knowledge state evolution for "②" and "⑤" within and between sessions.

According to Eqs. (21) and (22), the average difference of knowledge state evolution for "②" within the 9th and 10th sessions is 0.06 and 0.09, while that between the two sessions is 0.13. Similarly, the average difference of knowledge evolution for "⑤" within the 9th and 10th sessions is 0.02 and 0.07, while that between the two sessions is 0.3. It indicates that the knowledge states predicted by our ELPKT are consistent with the learners' responses within and between sessions. Moreover, the average difference in knowledge state evolution between sessions is greater than that within sessions, which validates our ELPKT can capture the knowledge state shifts between sessions.

The knowledge retention in inter-session modeling is effective. In Fig. 7, the learner's responses to knowledge concepts "③" and "④" are correct, and her knowledge level is always high in the adjacent 9th and 10th sessions. It indicates that the ELPKT model can consolidate well-mastered knowledge into long-term knowledge, making the responses to well-mastered knowledge consistent in adjacent sessions.

ELPKT models fine-grained learning behavior effectively. ELPKT models learners' guesses and engagement by considering fine-grained learning behaviors to assess their knowledge levels. The learner spent 249 s incorrectly answering exercise e_2, with a difficulty level 8. ELPKT may perceive that the learner's engagement on a difficult question is helpful for knowledge growth. Therefore, the knowledge state on "②" increases after solving e_2 in Fig. 7. The knowledge state will significantly decrease if the learner's behavior is deemed an obvious guess. For example, the learner requests 4 system hints within 35 s on solving e_{16}. ELPKT assumes that the learner guesses the answer, which results in a significant decrease in knowledge "②".

ELPKT can provide interpretability. ELPKT focuses on exercise difficulty level and learners' fine-grained learning behavior to measure learners' knowledge level, which provides interpretability for learners' knowledge level and the predicted performance. For example, in the 10th session, despite answering exercise e_{14} correctly, the learner's knowledge level on "⑤" is almost unchanged. By checking the learner's answering recording in Table 9, we find that the learner spent only 6 s on exercise e_{14}, with a difficulty level of 2. ELPKT may perceive that exercise e_{14} is particularly easy, resulting in little knowledge growth.

ELPKT can simulate knowledge forgetting and decay within and between sessions. As can be seen in Fig. 7, after interacting with exercise e_6 and e_{10}, the knowledge levels on "⑥" and "①" gradually decays over time. This indicates that ELPKT can gradually forget and decay the knowledge that was not applied during learning.

6.2.6. Discussion

While ELPKT enables capturing the knowledge state shifts between sessions and tracing learners' knowledge state effectively, we highlight several limitations that need to be considered. First, despite we set an interval threshold for session splitting by comprehensively analyzing the learning process and the hyperparameter experiments, it still has room for improvement. The sessions split by our method are unified based on all learning sequences, which may not satisfy personalized learning. Second, we validate the effectiveness of offline inter-session modeling through extensive experiments. Due to a lack of observable offline data, we consider short- to long-term knowledge retention and short-term knowledge decay in inter-session modeling. However, due to the diversity of learners' offline learning, it may not fully capture the knowledge state shifts between sessions when only considering knowledge retention and decay in offline time.

7. Conclusion

In this paper, we propose a method of enhancing learning process modeling for session-aware knowledge tracing, ELPKT, to effectively trace learners' knowledge state evolution in the learning process. Specifically, we conduct in-depth data analysis to understand the learners' learning process and their session-form learning pattern. Then, we validate the knowledge state shifts between sessions from the knowledge concept level through empirical study. Next, the ELPKT models the learning process as intra-sessions and inter-sessions at the knowledge concept level to track learners' knowledge state across sessions. Extensive experiments validate that the proposed ELPKT outperforms the existing methods in tracing learners' knowledge. Moreover, ELPKT can capture the knowledge state shifts between sessions effectively and provide interpretability for the predicted results. The method of quantifying the difference of knowledge state evolution within and between sessions proposed in this article can also be applied to measure users' fine-grained interest evolution in several downstream tasks, including learning resource recommendations or other interest-based recommenders.

In future work, several directions can be considered to enhance the learning process modeling. Firstly, developing an adaptive method to split sessions that satisfy personalized learning will be meaningful. Secondly, inspired by the benefits of uniform sequences in sequential recommendation [39], it will be interesting to explore data augmentation techniques to generate learning records for offline times. Lastly, to provide more valuable insights to learners and educators, it will be promising to explore the optimal learning patterns based on learners' learning time allocation.

CRediT authorship contribution statement

Chunli Huang: Writing – original draft, Software, Methodology, Conceptualization. **Wenjun Jiang:** Writing – original draft, Software, Methodology, Conceptualization. **Kenli Li:** Investigation, Formal analysis. **Jie Wu:** Writing – review & editing. **Ji Zhang:** Data curation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant No. 62172149) and the National Key R&D Program of China (Grant No. 2022ZD0118302).

Data availability

Data will be made available on request.

References

[1] S. Shen, Q. Liu, Z. Huang, Y. Zheng, M. Yin, M. Wang, E. Chen, A survey of knowledge tracing: Models, variants, and applications, IEEE Trans. Learn. Technol. (2024) 1–22.

[2] G. Abdelrahman, Q. Wang, B. Nunes, Knowledge tracing: A survey, ACM Comput. Surv. 55 (11) (2023).

[3] S. Shen, Z. Huang, Q. Liu, Y. Su, S. Wang, E. Chen, Assessing student's dynamic knowledge state by exploring the question difficulty effect, in: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2022, pp. 427–437.

[4] S. Minn, J.-J. Vie, K. Takeuchi, H. Kashima, F. Zhu, Interpretable knowledge tracing: Simple and efficient student modeling with causal relations, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 36, (11) 2022, pp. 12810–12818.

[5] S. Pandey, J. Srivastava, RKT: Relation-aware self-attention for knowledge tracing, in: Proceedings of the 29th ACM International Conference on Information & Knowledge Management, CIKM '20, ACM, New York, NY, USA, 2020, pp. 1205–1214.

[6] J. Chen, Z. Liu, S. Huang, Q. Liu, W. Luo, Improving interpretability of deep sequential knowledge tracing models with question-centric cognitive representations, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, (12) 2023, pp. 14196–14204.

[7] T. Long, Y. Liu, J. Shen, W. Zhang, Y. Yu, Tracing knowledge state with individual cognition and acquisition estimation, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '21, ACM, 2021, pp. 173–182.

[8] C. Wang, S. Sahebi, Continuous personalized knowledge tracing: Modeling long-term learning in online environments, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23, ACM, New York, NY, USA, 2023, pp. 2616–2625.

[9] K. Nagatani, Q. Zhang, M. Sato, Y.-Y. Chen, F. Chen, T. Ohkuma, Augmenting knowledge tracing by considering forgetting behavior, in: The World Wide Web Conference, WWW '19, ACM, New York, NY, USA, 2019, pp. 3101–3107.

[10] A. Ghosh, N. Heffernan, A.S. Lan, Context-aware attentive knowledge tracing, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '20, Association for Computing Machinery, 2020, pp. 2330–2339.

[11] S. Shen, Q. Liu, E. Chen, Z. Huang, W. Huang, Y. Yin, Y. Su, S. Wang, Learning process-consistent knowledge tracing, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 1452–1460.

[12] C. Wang, W. Ma, M. Zhang, C. Lv, F. Wan, H. Lin, T. Tang, Y. Liu, S. Ma, Temporal cross-effects in knowledge tracing, in: Proceedings of the 14th ACM International Conference on Web Search and Data Mining, WSDM '21, ACM, New York, NY, USA, 2021, pp. 517–525.

[13] B. Xu, Z. Huang, J. Liu, S. Shen, Q. Liu, E. Chen, J. Wu, S. Wang, Learning behavior-oriented knowledge tracing, in: Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2023, pp. 2789–2800.

[14] M. Zhang, X. Zhu, C. Zhang, W. Qian, F. Pan, H. Zhao, Counterfactual monotonic knowledge tracing for assessing students' dynamic mastery of knowledge concepts, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23, 2023, pp. 3236–3246.

[15] Y. Yin, L. Dai, Z. Huang, S. Shen, F. Wang, Q. Liu, E. Chen, X. Li, Tracing knowledge instead of patterns: Stable knowledge tracing with diagnostic transformer, in: Proceedings of the ACM Web Conference 2023, WWW '23, ACM, New York, NY, USA, 2023, pp. 855–864.

[16] S. Shen, E. Chen, B. Xu, Q. Liu, Z. Huang, L. Zhu, Y. Su, Quiz-based knowledge tracing, 2023, arXiv preprint arXiv:2304.02413.

[17] F. Ke, W. Wang, W. Tan, L. Du, Y. Jin, Y. Huang, H. Yin, Hitskt: A hierarchical transformer model for session-aware knowledge tracing, Knowl.-Based Syst. 284 (2024) 111300.

[18] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L.J. Guibas, J. Sohl-Dickstein, Deep knowledge tracing, Adv. Neural Inf. Process. Syst. 28 (2015).

[19] C.-K. Yeung, D.-Y. Yeung, Addressing two problems in deep knowledge tracing via prediction-consistent regularization, in: Proceedings of the Fifth Annual ACM Conference on Learning at Scale, in: L@S '18, ACM, New York, NY, USA, 2018.

[20] J. Zhang, X. Shi, I. King, D.-Y. Yeung, Dynamic key-value memory networks for knowledge tracing, in: Proceedings of the 26th International Conference on World Wide Web, WWW '17, International World Wide Web Conferences Steering Committee, 2017, pp. 765–774.

[21] G. Abdelrahman, Q. Wang, Knowledge tracing with sequential key-value memory networks, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 175–184.

[22] S. Pandey, G. Karypis, A self-attentive model for knowledge tracing, in: 12th International Conference on Educational Data Mining, EDM 2019, International Educational Data Mining Society, 2019, pp. 384–389.

[23] L. Zhang, X. Xiong, S. Zhao, A. Botelho, N.T. Heffernan, Incorporating rich features into deep knowledge tracing, in: Proceedings of the Fourth, 2017 ACM Conference on Learning @ Scale, in: L@S '17, ACM, New York, NY, USA, 2017, pp. 169–172.

[24] Y. Choi, Y. Lee, J. Cho, J. Baek, B. Kim, Y. Cha, D. Shin, C. Bae, J. Heo, Towards an appropriate query, key, and value computation for knowledge tracing, in: Proceedings of the Seventh ACM Conference on Learning @ Scale, in: L@S '20, ACM, 2020, pp. 341–344.

[25] S. Yang, X. Yu, Y. Tian, X. Yan, H. Ma, X. Zhang, Evolutionary neural architecture search for transformer in knowledge tracing, Adv. Neural Inf. Process. Syst. 36 (2024).

[26] M.J. Anzanello, F.S. Fogliatto, Learning curve models and applications: Literature review and research directions, Int. J. Ind. Ergon. 41 (5) (2011) 573–583.

[27] L. Averell, A. Heathcote, The form of the forgetting curve and the fate of memories, J. Math. Psychol. 55 (1) (2011) 25–35.

[28] H. Ebbinghaus, Memory: A contribution to experimental psychology, Ann. Neurosci. 20 (4) (2013) 155.

[29] Z. Huang, Q. Liu, Y. Chen, L. Wu, K. Xiao, E. Chen, H. Ma, G. Hu, Learning or forgetting? A dynamic approach for tracking the knowledge proficiency of students, ACM Trans. Inf. Syst. 38 (2) (2020) 1–33.

[30] D. Shin, Y. Shim, H. Yu, S. Lee, B. Kim, Y. Choi, SAINT+: Integrating temporal features for EdNet correctness prediction, in: LAK21: 11th International Learning Analytics and Knowledge Conference, in: LAK21, ACM, 2021, pp. 490–496.

[31] G. Abdelrahman, Q. Wang, Deep graph memory networks for forgetting-robust knowledge tracing, IEEE Trans. Knowl. Data Eng. 35 (8) (2023) 7844–7855.

[32] M. Chen, Q. Guan, Y. He, Z. He, L. Fang, W. Luo, Knowledge tracing model with learning and forgetting behavior, in: Proceedings of the 31st ACM International Conference on Information & Knowledge Management, CIKM '22, ACM, 2022, pp. 3863–3867.

[33] Y. Im, E. Choi, H. Kook, J. Lee, Forgetting-aware linear bias for attentive knowledge tracing, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23, ACM, 2023, pp. 3958–3962.

[34] J. Cui, Z. Chen, A. Zhou, J. Wang, W. Zhang, Fine-grained interaction modeling with multi-relational transformer for knowledge tracing, ACM Trans. Inf. Syst. 41 (4) (2023) 1–26.

[35] L. Wei, B. Li, Y. Li, Y. Zhu, Time interval aware self-attention approach for knowledge tracing, Comput. Electr. Eng. 102 (2022) 108179.

[36] M. Zhang, X. Zhu, C. Zhang, F. Pan, W. Qian, H. Zhao, No length left behind: Enhancing knowledge tracing for modeling sequences of excessive or insufficient lengths, in: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM '23, ACM, 2023, pp. 3226–3235.

[37] M. Zhang, X. Zhu, C. Zhang, Y. Ji, F. Pan, C. Yin, Multi-factors aware dual-attentional knowledge tracing, in: Proceedings of the 30th ACM International Conference on Information & Knowledge Management, CIKM '21, ACM, 2021, pp. 2588–2597.

[38] J. Yu, M. Lu, Q. Zhong, Z. Yao, S. Tu, Z. Liao, X. Li, M. Li, L. Hou, H.-T. Zheng, J. Li, J. Tang, MoocRadar: A fine-grained and multi-aspect knowledge repository for improving cognitive student modeling in MOOCs, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '23, ACM, 2023, pp. 2924–2934.

[39] Y. Dang, E. Yang, G. Guo, L. Jiang, X. Wang, X. Xu, Q. Sun, H. Liu, Uniform sequence better: Time interval aware data augmentation for sequential recommendation, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 37, (4) 2023, pp. 4225–4232.

[40] G.A. Miller, The magical number seven, plus or minus two: Some limits on our capacity for processing information, Psychol. Rev. 63 (2) (1956) 81.

[41] C.H. McGrath, B. Guerin, E. Harte, M. Frearson, C. Manville, Learning Gain in Higher Education, RAND Corporation, Santa Monica, CA, 2015.

[42] R.E. Slavin, Educational Psychology: Theory and Practice, Pearson, 2018.

[43] K.B. Lyle, C.R. Bego, R.F. Hopkins, J.L. Hieb, P.A. Ralston, How the amount and spacing of retrieval practice affect the short-and long-term retention of mathematics knowledge, Educ. Psychol. Rev. 32 (2020) 277–295.

[44] J.D. Karpicke, H.L. Roediger, Repeated retrieval during learning is the key to long-term retention, J. Mem. Lang. 57 (2) (2007) 151–162, http://dx.doi.org/10.1016/j.jml.2006.09.004, URL https://www.sciencedirect.com/science/article/pii/S0749596X06001367.

[45] G. Radvansky, Human Memory (4th ed.), fourth ed., Routledge, New York, NY, 2021.

[46] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[47] D.P. Kingma, J. Ba, Adam: A method for stochastic optimization, 2014, arXiv preprint arXiv:1412.6980.