frontiers | Frontiers in Psychology

METHODS
published: 01 August 2022
doi: 10.3389/fpsyg.2022.846621

A Knowledge Query Network Model Based on Rasch Model Embedding for Personalized Online Learning

Yan Cheng[1,2], Gang Wu[1], Haifeng Zou[1], Pin Luo[1] and Zhuang Cai[1]*

[1] School of Computer and Information Engineering, Jiangxi Normal University, Nanchang, China, [2] Jiangxi Provincial Key Laboratory of Intelligent Education, Nanchang, China

The vigorous development of online education has produced massive amounts of education data. How to mine and analyze education big data has become an urgent problem in the field of education and big data knowledge engineering. As for the dynamic learning data, knowledge tracing aims to track learners' knowledge status over time by analyzing the learners' exercise data, so as to predict their performance in the next time step. Deep learning knowledge tracking performs well, but they mainly model the knowledge components while ignoring the personalized information of questions and learners, and provide limited interpretability in the interaction between learners' knowledge status and questions. A context-aware attentive knowledge query network (CAKQN) model is proposed in this paper, which combines flexible neural network models with interpretable model components inspired by psychometric theory. We use the Rasch model to regularize the embedding of questions and learners' interaction tuples, and obtain personalized representations from them. In addition, the long-term short-term memory network and monotonic attention mechanism are used to mine the contextual information of learner interaction sequences and question sequences. It can not only retain the ability to model sequences, but also use the monotonic attention mechanism with exponential decay term to extract the hidden forgetting behavior and other characteristics of learners in the learning process. Finally, the vector dot product is used to simulate the interaction between the learners' knowledge state and questions to improve the interpretability. A series of experimental results on 4 real-world online learning datasets show that CAKQN has the best performance, and its AUC value is improved by an average of 2.945% compared with the existing optimal model. Furthermore, the CAKQN proposed in this paper can not only track learners' knowledge status like other models but also model learners' forgetting behavior. In the future, our research will have high application value in the realization of personalized learning strategies, teaching interventions, and resource recommendations for intelligent online education platforms.

Keywords: personalized education, deep learning, knowledge tracking, forgetting behavior, interpretability

OPEN ACCESS

Edited by:
Manuel Gentile, Istituto per le Tecnologie Didattiche (ITD) (CNR), Italy

Reviewed by:
Sílvio Manuel da Rocha Brito, Instituto Politécnico de Tomar (IPT), Portugal
David Paulo Ramalheira Catela, Polytechnic Institute of Santarém, Portugal

****Correspondence:***
Yan Cheng
chyan88888@jxnu.edu.cn

Specialty section:
This article was submitted to Cognitive Science, a section of the journal Frontiers in Psychology

Received: *31 December 2021*
Accepted: *23 May 2022*
Published: *01 August 2022*

Citation:
Cheng Y, Wu G, Zou H, Luo P and Cai Z (2022) A Knowledge Query Network Model Based on Rasch Model Embedding for Personalized Online Learning. Front. Psychol. 13:846621. doi: 10.3389/fpsyg.2022.846621

INTRODUCTION

With the rapid development of Internet technology and artificial intelligence technology in the field of education, online learning platforms such as massive open online courses (MOOCs) have become increasingly popular. Learners' activities on online learning platforms have generated massive amounts of educational data. How to mine and analyze large amounts of educational data has become an urgent problem in the field of education and big data knowledge engineering (Hu et al., 2020). Since learners' behavior, knowledge state, and psychological factors in the learning process are the key factors for evaluating their learning effectiveness (Yang and Li, 2018), and these factors are constantly changing over time, it is of great significance to construct a learner model oriented to dynamic learning data.

Different from the cognitive diagnosis model (CDM) for static learning data, knowledge tracing (KT) aims to dynamically track learners' knowledge status over time by analyzing the learners' historical exercise data, so as to predict their performance in the next time step. The learner's historical exercise data is a sequence composed of the questions, the knowledge components (KCs) contained in the questions, and the learner's answers (Liu et al., 2021). The three core elements of questions, KCs and learners constitute the three basic objects of the KT data processing, the interaction between them is shown in **Figure 1**. KT is the quantitative analysis and modeling of the relationship between three types of objects. For example, the prediction of students' knowledge mastery state is to calculate the mastery probability between "students and knowledge" by using the interaction between "students and problems" and the correlation information between "problems and knowledge." The interaction between different objects is the main information used in the KT modeling process (Sun et al., 2021). Therefore, the KT model not only needs to accurately assess the learner's knowledge state and predict their answer in the future but also needs to provide explanations for the interaction between different objects (Hu et al., 2020).

Traditional KT methods mainly include Bayesian knowledge tracking (BKT) (Corbett and Anderson, 1994) based on hidden Markov model (HMM) (Rabiner and Juang, 1986) and item response theory (IRT) (Fan, 1998). In recent years, researchers have tended to use more complex and flexible models like deep networks to make full use of hidden information in large-scale learner response datasets. The deep knowledge tracing (DKT) (Piech et al., 2015) model introduced recurrent neural network (RNN) into the KT field for the first time and achieved success. Compared with the traditional KT model, the predictive ability based on the deep learning method has been significantly improved. However, most of the current KT methods based on deep learning mostly use KCs to index questions, ignoring the rich information contained in the questions and the context. For example, investigating different questions of the same KC may cause individual differences between questions due to different difficulty settings. In addition, the personalized interaction between learners' knowledge status and questions representation is often overlooked, which leads to poor interpretability of the KT method based on deep learning. In response to the above problems, we propose a context-aware attentive knowledge query network (CAKQN) model based on the embedded Rasch model, which is the single parameter IRT model. First, input the learner interaction tuple and questions into the embedded component based on the Rasch model to obtain personalized representations of the learner interaction tuple and questions, and capture the characteristics of individual differences between different questions containing the same KC and the learners' personal abilities. Next, based on the definition of memory trace decline in educational psychology theory (Bailey, 1989) that human memory fades automatically over time, a network structure of long short-term memory network + monotonic attention mechanism is designed to learn personalized learner knowledge state and context-aware representation of the questions. The learning process of learners is continuous, so the sequence structure of learning records cannot be destroyed in the KT modeling process. The structure we designed uses a monotonic attention mechanism with an exponential decay term to reduce the importance of learner interaction tuples in the distant past without destroying the sequence structure of the learners' historical learning records, and it can extract features such as forgetting behavior that exist in the learning process of learners. Finally, based on the fact that learners answer questions based on their knowledge status and personal abilities, the vector dot product is used to simulate the personalized interaction between learners' knowledge status and questions to improve the interpretability of the model. We used four publicly available real online education datasets to evaluate the model. Experiments show that the CAKQN model has the best performance, and its AUC value is 2.945% higher than the existing optimal model on average. In addition, our paper also conducted a series of ablation analysis and knowledge tracking visualization experiments to verify the excellent interpretability and personalization capabilities of the CAKQN model. In the future, our research will have high application value in the realization of personalized learning strategies, teaching interventions, and resource recommendations for intelligent online education platforms.

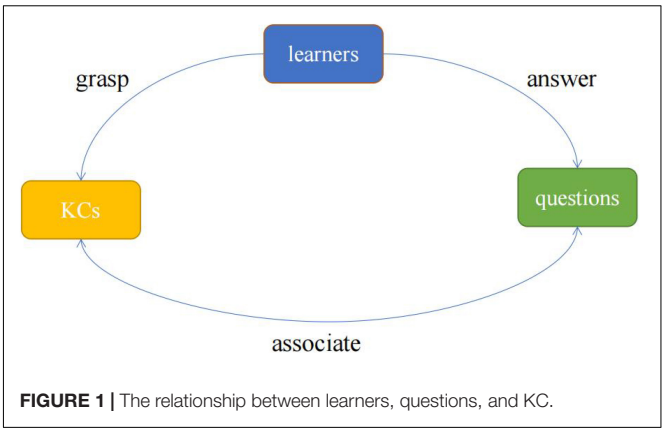

FIGURE 1 | The relationship between learners, questions, and KC.

RELATED WORK

Traditional Knowledge Tracking Methods

Traditional knowledge tracking methods are mainly divided into two categories: IRT and BKT, and IRT is one of the important psychological and educational theories (Cheng et al., 2019). The single-parameter IRT model (i.e., Rasch model) outputs the probability of learners answering the items correctly during the test according to the learner's ability level and the difficulty level of the items (i.e., questions). The probability is defined by the item response function with the following characteristics: if the learner's ability level is higher, the learner has a higher probability of answering an item correctly. Conversely, if an item is more difficult, the probability of the learner answering the item correctly is lower. The item response function is defined as follows:

$$P(a) = \sigma\left(\theta - \beta_j\right) = \frac{1}{1 + e^{-D(\theta - \beta_j)}} \tag{1}$$

The more complex two-parameter item response function introduces item discrimination α_j, which is defined as follows:

$$P(a) = \sigma\left(\theta - \beta_j\right) = \frac{1}{1 + e^{-D\alpha_j(\theta - \beta_j)}} \tag{2}$$

Where σ is the sigmoid function, D is a constant, usually set to 1.7, θ is learner's ability level, β_j is the difficulty level of item j. Since the IRT model was originally designed for educational testing environments, the model assumes that learners' abilities remain unchanged during the testing process. In reality, the knowledge state of learners changes with time step, so it cannot be directly applied to KT tasks.

The BKT model updates the learner's knowledge state through HMM modeling, and predicts the learner's performance at the next time step accordingly. However, many simplified assumptions used in the BKT model are impractical. One of them is that all learners and questions containing the same KC are considered the same. Therefore, the researchers studied various personalizations of the BKT model. Some researchers endow the BKT model with personalized capabilities on specific parameters of KC (Pardos and Heffernan, 2011) and specific parameters of learners (Yudelson et al., 2013). Some other researchers have also studied the synthesis of the BKT model and the IRT model (Khajah et al., 2014; Wilson et al., 2016) to enhance the model's personalization ability when dealing with questions and learners. However, such expansion usually requires a lot of feature engineering work and will result in a significant increase in computing requirements.

Deep Learning Knowledge Tracking

In recent years, deep learning has attracted attention from researchers with its powerful feature extraction capabilities. Many researchers have applied it to the KT field, which is called DLKT (deep learning knowledge tracing) (Liu et al., 2021). Compared with BKT and IRT, DLKT does not require manually annotated KC information and can capture more complex learner knowledge representations from large-scale learner response datasets. DKT and dynamic key-value memory network (DKVMN) (Zhang et al., 2017) have shown strong predictive ability in predicting learners' future performance, and have become the benchmark for subsequent DLKT methods. DKT takes the learner's historical learning interaction sequence as input, then uses RNN to encode it into the learner's knowledge state, and finally inputs it into a linear layer activated by a Sigmoid function to get the prediction result. DKT, which simply represents the learner's knowledge state as a vector, while DKVMN uses a static external matrix to store KC and uses a dynamic matrix to update the learner's mastery of KC. However, the simple splicing between the two vectors representing the learner's knowledge state and KC in the DKVMN model is not enough to explain the process of interaction between the learner's knowledge state and the KC contained in the question (Daniluk et al., 2017). The knowledge query network (KQN) (Lee and Yeung, 2019) model uses the vector dot product to more accurately simulate the interaction between the learner's knowledge state and KC, and achieves better results. Self-attentive knowledge tracing (SAKT) (Pandey and Karypis, 2019) model is the first to use the Transformer structure in the KT field to replace RNN to automatically focus on the record of questions in the learner's historical interaction sequence that has a greater impact on the prediction results and achieves model performance. The substantial increase. However, the above models use KCs to index questions, that is, all different questions containing the same KC are regarded as equivalent. This way ignores the rich information contained in the question itself and the context. Context-aware attention knowledge tracing (AKT) (Ghosh et al., 2020). The framework based on the SAKT model uses the Rasch model to regularize concept and question embeddings. These embeddings can capture questions that contain the same KC, without using too many parameters. In addition, AKT also uses a new monotonic attention mechanism to link learners' future responses to questions with their historical interaction sequences to extract features such as hidden forgetting behavior in the learning process of learners. However, the AKT model also uses unreasonable vector simple splicing to simulate learner knowledge status and question interaction, and it loses the ability to model sequence due to the Transformer structure like SAKT.

Considering the advantages and disadvantages of KQN model and AKT model, this paper proposes a context-aware knowledge query network (CAKQN) based on Rasch model embedding. It not only retains the ability of model sequence but also obtains personalized contextual representations of questions and learners. We improve the model's performance in predicting future learner responses. Moreover, the interpretability of the model in terms of learner knowledge status and questions interaction is enhanced.

OUR PROPOSED METHOD

This section first introduces the problem setup of knowledge tracing and the symbolic representation of related concepts, then introduces the difference between ordinary attention mechanism and monotonic attention mechanism with exponential decay, and then describes the overall context-aware knowledge query

network model based on Rasch model embedding framework, and finally introduce each component of the model and its loss function in turn.

Knowledge Tracing Problem Setup

Assuming that there are M questions and N KCs in the original dataset, each learner's interaction record is composed of the learner's long questions and responses at each time step. For the learner i at time step t, a learner interaction tuple $x_t = (q_t^i, c_t^i, r_t^i)$ is composed of: the question q_t^i he or she answered, the KC c_t^i covered by the question, and the learner's response r_t^i to the question. Where q_t^i is the question index, $q_t^i \in \{1, \cdots, M\}$, c_t^i is the KC index, $c_t^i \in \{1, \cdots, N\}$, and r_t^i is the response, $r_t^i = \{0, 1\}$. Under this notation, $(q_t, c_t, 1)$ means learner i responded to question q_t on concept c_t correctly at time step t. This setting is different from some previous deep knowledge tracking work, which often ignores the question index and set the learner's interaction tuple as (c_t^i, r_t^i). For convenience, the superscript i is omitted in the following discussion. Therefore, given learner's historical learning interaction sequence $X_t = \{x_1, x_2, \cdots, x_t\}$ at time step t and question q_{t+1} on concept c_{t+1} at time step $t+1$, the goal of the KT model is to find the probability $P(r_{t+1} = 1|X_t, q_{t+1}, c_{t+1})$.

Monotonic Attention Mechanism With Exponential Decay

Under the ordinary dot product attention mechanism, the input is mapped to three vectors: *Query*, *Key*, and *Value* by embedding layer, and values of dimension $D_q = D_k$, D_k and D_v. Let $q_t \in \mathbb{R}^{D_k \times 1}$ donate the *Query* corresponding at time step t, the calculation formula of the scaled dot product attention value $\alpha_{t,\tau}$ normalized by the softmax function is:

$$\alpha_{t,\tau} = Softmax(\frac{q_t^T k_\tau}{\sqrt{D_k}}) = \frac{exp(\frac{q_t^T k_\tau}{\sqrt{D_k}})}{\sum_{\tau'} exp(\frac{q_t^T k_\tau}{\sqrt{D_k}})} \in [0, 1] \quad (3)$$

Where $k_\tau \in \mathbb{R}^{D_k \times 1}$ donate *Key* at time step τ .

However, this ordinary zoom dot product attention mechanism is not enough for KT tasks. The reason is that learners have forgetting behaviors in the learning process, and learners will have memory decline in the real world (Pashler et al., 2009). In other words, when the model predicts the learner's reaction to the next question, his performance in the distant past is not as important as his recent performance. Therefore, Ghosh et al. (2020) add a multiplicative exponential decay term to the attention scores. So the calculation of the new monotonic attention mechanism is as follows:

$$\alpha'_{t,\tau} = \frac{exp(s_{t,\tau})}{\sum_{\tau'} exp(s_{t,\tau'})} \quad (4)$$

$$s_{t,\tau} = \frac{exp(-\theta \cdot d(t, \tau)) \cdot q_t^T k_\tau}{\sqrt{D_k}} \quad (5)$$

Where $\theta > 0$ is a learnable decay rate parameter, and $d(t, \tau)$ is temporal distance measure between time steps t and τ. In other words, the attention weight of the current question to the past question not only depends on the similarity between the corresponding sums, but also depends on the relative time steps between them. The calculation method of $d(t, \tau)$ is as follows:

$$d(t, \tau) = |t - \tau| \cdot \sum_{t'=\tau+1}^{t} \gamma(t, t') \quad (6)$$

$$\gamma(t, t') = \frac{exp(\frac{q_t^T k_{t'}}{\sqrt{D_k}})}{\sum_{1 \le \tau' \le t} exp(\frac{q_t^T k_{\tau'}}{\sqrt{D_k}})} \quad (7)$$

The calculation formula of the final output of the monotonic attention mechanism is as follows:

$$Monotonic\_Attention(Query, Key, Value) = \sum_{\tau=1}^{t} \alpha'_{t,\tau} v_\tau \quad (8)$$

Where $v_\tau \in \mathbb{R}^{D_k \times 1}$ donate *Key* at time step τ .

Model Framework

This paper proposes a context-aware knowledge query network based on Rasch model embedding. **Figure 2** shows the overall framework of the model. It contains 4 components: *Embedded Layer Based on Rasch Model*, *Knowledge State Encoder*, *Question Encoder*, and *Knowledge Status Query*.

(1) *Embedded Layer Based on Rasch Model*: Get the personalized embedding of the learner interaction tuple at the current time step and the next-time step question, and capture the characteristics of individual differences between different questions on the same KC and the learners' personal abilities.

(2) *Knowledge State Encoder*: First, use the location information provided by the long short-term memory network to model the context of the learner's historical interaction sequence, and retain the ability of the model to model the sequence. Then, the monotonic attention mechanism with exponential decay term is used to reduce the importance of learner interaction tuples in the distant past, extract the forgetting behavior and other characteristics of learners in the learning process, and obtain the contextual perception vector of the learner's knowledge state at the current time step.

(3) *Question Encoder*: It is exactly the same as the network structure adopted by the knowledge state encoder to obtain the context awareness vector of the question at the current time step.

(4) *Knowledge Status Query*: The dot product operation is performed on the vector representing the learner's knowledge state and the question at the current time step to simulate the interaction between the learner's knowledge state and the question, and the result of the dot product is input into the sigmoid function to obtain the final prediction of the probability that the learner will answer correctly at the next time step.

Embedding Layer Based on Rasch Model

Existing KT methods mostly use KC to index questions, that is, set $q_t = c_t$, because the number of questions in the real world is far greater than the number of KC, so using KC to index

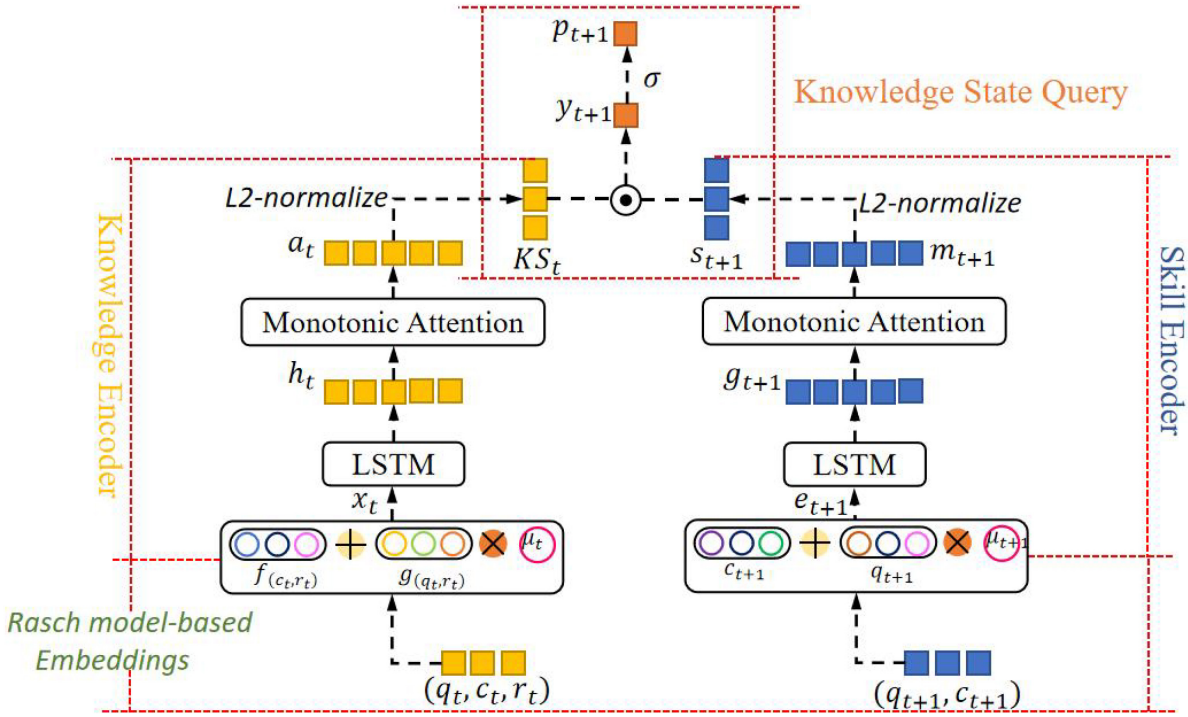

FIGURE 2 | The overall framework of the CAKQN model.

questions can effectively avoid over-parameterization and over-fitting. However, this setting ignores the individual differences between question covering the same KC, and limits the flexibility of the KT method and its ability to be personalized.

This article uses the classic Rasch model in psychometric theory to construct learner interaction tuples and question embedding. There are two important parameters in the Rasch model: the difficulty of the question and the ability of the learners. Therefore, at time step t, the final embedded representation of the learner's interaction tuple is expanded to:

$$x_t = f_{(c_t,r_t)} + \mu_t \cdot g_{(q_t,r_t)} \tag{9}$$

Where $f_{(c_t,r_t)} \in \mathbb{R}^D$, $g_{(q_t,r_t)} \in \mathbb{R}^D$, they respectively, represent the embedding vector of the KC response tuple and the embedding vector of the question response tuple. And μ_t is a learnable scalar, which represents the learner's ability parameter. At the next time step, the final embedded representation of the question is expanded to:

$$e_{t+1} = c_{t+1} + \mu_{t+1} \cdot q_{t+1} \tag{10}$$

Where $c_{t+1} \in \mathbb{R}^D$ is the embedding vector of KC contained in this question, $q_{t+1} \in \mathbb{R}^D$ is the embedding vector of the question. And μ_{t+1} is also a learnable scalar, it represents the difficulty parameter, which controls the degree of deviation of the question from the KC contained in it. These Rasch model-based embeddings strike an appropriate balance between obtaining personalized representations and avoiding excessive parameterization.

Knowledge State Encoder

In the *Knowledge State Encoder*, the structure of the LSTM layer + monotonic attention mechanism layer is used to obtain the context perception results of learner interaction sequences. The way learners understand and learn when answering questions is based on their own knowledge state, and the learner's knowledge state is related to the learner's historical learning interaction sequence. For two learners with different historical learning interaction sequences, the way they understand the same question and the knowledge they gain from the exercise may be different. Therefore, we use the LSTM structure to ensure that the original learner history learning interaction sequence is not destroyed on the time scale, and introduce the monotonic attention mechanism to summarize the performance of the past learners in the correct time range, tap the hidden features of the learning process, and then obtain their knowledge state. Given input x_t, the knowledge state encoder first inputs it to the LSTM layer to obtain its hidden state h_t. Then input h_t to the monotonic attention mechanism layer to get the weighted vector a_t, and finally a through a fully connected layer and L2 normalization to get the final output knowledge state vector KS_t. The calculation process is as follows:

$$\begin{cases} i_t = \sigma\left(W_i \left[x_t,\ h_{t-1},\ c_{t-1}\right] + b_i\right) \\ f_t = \sigma(W_f[x_t,\ h_{t-1},\ c_{t-1}] + b_f) \\ o_t = \sigma(W_o[x_t,\ h_{t-1},\ c_{t-1}] + b_o) \\ c_t = f_t c_{t-1} + i_t \tanh\left(W_c \left[x_t,\ h_{t-1}\right] + b_c\right) \\ h_t = o_t tanh(c_t) \end{cases} \tag{11}$$

Where i_t, f_t, o_t, c_t are the input gate, forget gate, output gate and unit state, respectively.

$$a_t = Monotonic\_Attention(x_t, x_t, h_t) \tag{12}$$

$$KS_t = L2\_normalize(W_{h,KS}a_t + b_{h,KS}) \tag{13}$$

Where $W_{h,KS} \in \mathbb{R}^{d \times H_{LSTM}}$, $b_{h,KS} \in \mathbb{R}^d$, and H_{LSTM} is the size of the hidden layer of the LSTM, d is the dimension of the knowledge state vector KS_t and the question vector S_{t+1}. *L2_normalize*is L2 normalization, the reason for this limitation is to allow the knowledge state vector and the question vector to be a dot product. In addition, in order to avoid overfitting, regularization is used in the output layer of LSTM.

Question Encoder

In this article, the question encoder uses the same network structure as the knowledge state encoder, and the purpose is also to capture the context-aware results of the question at the next time step. The specific calculation process of the input question embedding e_{t+1} to obtain the question vector s_{t+1} by the question encoder is as follows:

$$\begin{cases} i_t = \sigma\left(W_i\ \left[e_{t+1},\ g_{t-1},\ c_{t-1}\right] + b_i\right) \\ f_t = \sigma(W_f[e_{t+1},\ g_{t-1},\ c_{t-1}] + b_f) \\ o_t = \sigma(W_o[e_{t+1},\ g_{t-1},\ c_{t-1}] + b_o) \\ c_t = f_t c_{t-1} + i_t \tanh\left(W_c\ \left[e_{t+1},\ g_{t-1}\right] + b_c\right) \\ g_{t+1} = o_t tanh(c_t) \end{cases} \tag{14}$$

$$m_{t+1} = Monotonic\_Attention(e_{t+1}, e_{t+1}, g_{t+1}) \tag{15}$$

$$s_{t+1} = L2\_normalize(W_{h,KS}m_{t+1} + b_{h,KS}) \tag{16}$$

Knowledge Status Query

Do the dot product operation on the dimensional knowledge state vector KS_t and the dimensional question vector S_{t+1} obtained by the knowledge state encoder and the item encoder, respectively, and then input the result into the sigmoid activation function to obtain the final prediction of the probability p_{t+1} that the learner answers the next question correctly. Calculated as follows:

$$y_{t+1} = KS_t \cdot S_{t+1} \tag{17}$$

$$p_{t+1} = \sigma(y_{t+1}) \tag{18}$$

The dot product of knowledge state vector and question vector conforms to the process of real world middle school learners answering questions based on their own knowledge state (Lee and Yeung, 2019), which makes the model more explanatory.

Optimization

We use the backpropagation algorithm to train the network model, and update the model parameters by minimizing the cross entropy loss of the prediction probability and the labeled result. At each time step t, calculate the cross entropy loss result of a single learner, and sum the $t = 1, \cdots, T-1$ loss of all learners to get the total loss. The specific calculation process is:

$$\ell(\theta_{model}|r^i_{t+1}, p^i_{t+1}) = -[r^i_{t+1} log p^i_{t+1} + (1 - r^i_{t+1}) log(1 - p^i_{t+1})] \tag{19}$$

$$\mathcal{L}(\theta_{model}|r_{2:t+1}, p_{2:t+1}) = \sum_i \sum_{t=1}^{T-1} \ell(\theta_{model}|r^i_{t+1}, p^i_{t+1}) \tag{20}$$

TABLE 1 | Statistics of dataset.

Dataset	learners	KCs	Questions	Responses
ASSISTments2009	4,151	110	16,891	325,637
ASSISTments2015	19,840	100	–	683,801
ASSISTments2017	1,709	102	3,162	942,816
Statics2011	333	1,223	–	189,297

EXPERIMENTS

In this section, we first introduce the details of the dataset, experimental parameter settings and evaluation indicators, and then show the performance of this model and other models in 4 real-world online education datasets. Finally, we use ablation experiments to further verify the effectiveness of the Rasch model-based embedding, monotonic attention mechanism and question context-aware representation.

Datasets

We used four publicly available real online education datasets to evaluate the model, namely ASSISTments2009, ASSISTments2015, ASSISTments2017[1], and Statics2011[2]. The ASSISTments datasets are collected from the ASSISTments online tutoring platform. And the ASSISTments2009 dataset has been the accepted standard dataset of the KT method for the past 10 years. The Statics2011 dataset was collected from a university-level statics engineering course. In all datasets, the preprocessing steps in this paper follow a series of standards in Ghosh et al. (2020). In **Table 1**, we list the number of learners, KCs (i.e., concepts, knowledge points), questions, and learner interaction tuples. In these datasets, only the ASSISTments2009 and ASSISTments2017 datasets contain question IDs. Therefore, the model based on the Rasch model embedding is only applicable to these two datasets.

Experimental Setup and Evaluation Index

We use the five-fold cross-validation method to start the experiment based on PyTorch version 1.2.0. The division of all datasets is consistent with Ghosh et al. (2020), 20% is used as the test set, 20% is used as the validation set, and 60% is used as the training set. And we use the grid search method

[1]The ASSISTments datasets are retrieved from https://sites.google.com/site/assistmentsdata/home and https://sites.google.com/view/assistmentsdatamining/.

[2]The Statics2011 dataset is retrieved from https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=507.

TABLE 2 | The predicted results of different methods on knowledge tracing.

Model	AUC (%)			
	ASSISTments2009	**ASSISTments2015**	**ASSISTments2017**	**Statics2011**
IRT+	77.40*	–	–	–
BKT+	69*	–		75*
DKT	80.53 ± 0.2*	72.52 ± 0.1*	72.63 ± 0.1*	80.20 ± 0.2*
DKVMN	81.57 ± 0.1*	72.68 ± 0.1*	70.73 ± 0.1*	82.84 ± 0.1*
KQN	82.32 ± 0.05*	73.40 ± 0.02*	73.33 ± 0.03*	83.20 ± 0.05*
SAKT	84.8*	85.4*	72.12*	85.3*
AKT-NR	81.69 ± 0.004*	78.28 ± 0.002*	72.82 ± 0.003*	82.65 ± 0.004*
AKT-R	83.46 ± 0.003*	–	77.02 ± 0.002*	–
CAKQN-R	**87.04 ± 0.004**	–	**79.33 ± 0.002**	–
CAKQN-NR	85.54 ± 0.003	**88.88 ± 0.004**	76.45 ± 0.003	**85.43 ± 0.001**

*The symbol * means the result is from other paper. The best results are shown in bold.*

on the validation set to determine the optimal parameters. We use $\{10^{-6}, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$, {64, 128, 256, 512}, {64, 128, 256, 512}, {0, 0.05, 0.1, 0.15, 0.2, 0.25}, and {32, 64, 128, 256, 512} as values of the learning rate, the input embedding dimension, the hidden state dimension of LSTM, the dropout rate for the LSTM network, and the dimension of knowledge state vector and question vector, respectively. Finally, we set the maximum number of epochs to 300, the default optimizer to Adam, the learning rate to 10^{-4}, batch size to 32, the input embedding dimension to 128, the dimension of the LSTM hidden layer to 128, the dropout rate to 0.1, the dimension of knowledge state vector and question vector to 128.

With reference to most of the KT research work, we use the area under the curve (AUC) as an evaluation model to predict the performance of the learner's next interaction. The higher the AUC, the better the model's predictive performance.

Experimental Results and Analysis

Comparative Experiment

On four educational datasets, the CAKQN model proposed in our paper is compared with several common traditional network KT model representatives including IRT+ (Pardos and Heffernan, 2011), BKT+ (Yudelson et al., 2013) and neural network representative baseline models, including DKT (Piech et al., 2015), DKVMN (Zhang et al., 2017), KQN (Lee and Yeung, 2019), SAKT (Pandey and Karypis, 2019), AKT (Ghosh et al., 2020), the experimental results are shown in **Table 2**. Note that best models are bold, the results with * are form other paper.

Table 2 lists the performance of all KT methods across all datasets for predicting future learner responses. CAKQN-R and CAKQN-NR represent variants of the CAKQN model with and without the embedding based on the Rasch model, respectively. Similarly, AKT-R and AKT-NR represent variants of the AKT model with and without the embedded Rasch model in Ghosh et al. (2020), respectively. The experimental results show that the CAKQN-R model proposed in this paper is better than the existing model, and its AUC value is 2.945% higher than the existing optimal model AKT-R on average. Note that IRT+ and BKT+ have the lowest prediction performance on the four datasets compared to the neural network representing the four datasets. This indicates that both methods rely on experts to label KC, and the model cannot capture more information like deep neural networks. In the DLKT model, the average prediction performance of the KQN model on the four datasets is significantly improved compared to DKT and DKVMAN. This is because the KQN model is more explanatory in terms of learner knowledge interaction. And CAKQN-R and CAKQN-NR, which also use dot products to represent the interaction process between learner knowledge and questions, have achieved better performance on all datasets. This is related to its different network structure, the monotonic attention mechanism introduced and the embedding based on the Rasch model. Taking a closer look, the SAKT, AKT, and CAKQN models that introduce the attention mechanism and its variants have achieved better results than the general DLKT models such as DKT, DKVMN, and KQN. Because the attention mechanism can link the KC at the next time step with the related KC in the learner's past interaction sequence, the DLKT model with the attention mechanism can more accurately describe the knowledge state of each learner, thereby improving the performance of the model. Among them, the CAKQN-R model achieved better results than other DLKT models with attention mechanisms on the two ASSISTments datasets with question IDs. This proves that the CAKQN-R model can dig more complex features such as forgetting behavior in learner interaction sequences, obtain more accurate learner knowledge status and improve the prediction effect. Comparing the CAKQN-NR and AKT-NR models with the same monotonic attention mechanism, CAKQN-NR model proposed in this paper uses the network structure of LSTM+monotonic attention mechanism to retain the ability of the model to model the sequence, which can not only ensure that the original learner's historical learning interaction sequence is not damaged on the time scale, but also extract complex features of learners such as forgetting behavior. More importantly, it also provides a more interpretable interaction process between learner knowledge and questions, which contributes to a better prediction effect than AKT-R.

TABLE 3 | Experimental comparison between CAKQN and variant that do not use contextual aware question and response representations.

Model	AUC (%)			
	ASSISTments2009	**ASSISTments2015**	**ASSISTments2017**	**Statics2011**
CAKQN^{raw}-NR	84.49 ± 0.004	85.31 ± 0.004	74.84 ± 0.002	85.13 ± 0.001
CAKQN-NR	**85.54 ± 0.003**	**88.88 ± 0.004**	**76.45 ± 0.003**	**85.43 ± 0.001**
CAKQN^{raw}-R	86.12 ± 0.004	–	77.14 ± 0.003	–
CAKQN-R	**87.04 ± 0.004**	–	**79.33 ± 0.002**	–

The best results are shown in bold.

TABLE 4 | Experimental comparison between CAKQN and variants with other attention mechanism.

Model	AUC (%)			
	ASSISTments2009	**ASSISTments2015**	**ASSISTments2017**	**Statics2011**
SAKT	84.8*	85.4*	72.12*	85.3*
CAKQN-NR^{nl}	84.01 ± 0.005	80.52 ± 0.011	71.84 ± 0.004	83.89 ± 0.001
CAKQN-NR	**85.54 ± 0.003**	**88.88 ± 0.004**	**76.45 ± 0.003**	**85.43 ± 0.001**
CAKQN-R^{nl}	85.52 ± 0.004	–	75.44 ± 0.003	–
CAKQN-R	**87.04 ± 0.004**	–	**79.33 ± 0.002**	–

*The symbol * means the result is from other paper. The best results are shown in bold.*

Finally, comparing CAKQN-R and CAKQN-NR, we found that CAKQN-R has better prediction performance on both datasets. This proves that the embedding based on the Rasch model can capture the characteristics of individual differences between different questions of the same KC and the personal abilities of learners, and obtain more accurate personalized representations of learner interaction tuples and questions, thereby improving the performance of the model.

Ablation Experiment

In order to further verify the three key innovations in the CAKQN model: context-aware representation of question vectors, monotonic attention mechanism, and embedding based on the Rasch model, three additional ablation experiments were carried out in this paper. The first experiment is the comparison of CAKQN-R, CAKQN-NR and its variants CAKQN^{raw}-R and CAKQN^{raw}-NR. The structure of CAKQN^{raw}-R and CAKQN^{raw}-NR Question Encoder is the same as the KQN model. It uses a multi-layer perceptron (MLP) to directly input the question embedding to obtain the question vector, the number of hidden layers is 1 and the dimension is 128. The second experiment is to compare CAKQN-R, CAKQN-NR, SAKT models and two variants CAKQN-R^{nl} and CAKQN-NR^{nl} without monotonic attention mechanism. The tow variants use ordinary dot product attention to capture the time dependence in the learner's response data. The last one is the experiment is a comparison between CAKQN-R and variant CAKQN-IRT. The CAKQN-IRT model is based on the DIRT framework proposed in Cheng et al. (2019). Specifically, the *Knowledge State Encoder* and *Question Encoder* components used in the CAKQN-IRT model are the same as CAKQN-R, but the difference is that CAKQN-IRT uses direct embedding instead of Rasch embedding. The *Knowledge State Encoder* component of CAKQN-IRT obtains the learners' ability θ, one *Question Encoder* component inputs the question and KC embedding to obtain the distinction of the question α_j, and the other exactly the same *Question Encoder* component inputs the question embedding to obtain the difficulty of the question β_j. Finally, the obtained parameters are substituted into the two-parameter IRT model formula in section "Traditional Knowledge Tracking Methods" for prediction.

Table 3 shows the results of the first ablation experiment based on the context-aware representation of the question vector. In all datasets, CAKQN-R and CAKQN-NR are better than CAKQN^{raw}-R and CAKQN^{raw}-NR. These results show that our context-aware representation of the question is effective in summarizing the relationship between the question at the next time step and the historical question.

Table 4 shows the results of the second ablation experiment of the monotonic attention mechanism. On all datasets, CAKQN-NR is significantly better than other attention mechanisms, including SAKT. In the case of both using Rasch-based model embedding, CAKQN-R still achieves better results than CAKQN-R^{nl} on the two datasets. The reason for this is that it is different from the common language tasks with strong long-distance dependence between words. The dependence of future learner performance on the past is restricted to a much shorter time window for their forgetting behaviors. Therefore, the monotonic attention mechanism with exponential decay when calculating the attention weight can effectively capture the short-term dependence on the past on the time scale to simulate the forgetting behavior of learners in the learning process.

Table 5 shows the results of the third ablation experiment based on the embedding of the Rasch model. Both models are only tested on the two ASSISTments datasets where the question ID in the dataset is available. On these two datasets, CAKQN-R is significantly better than CAKQN-IRT in the

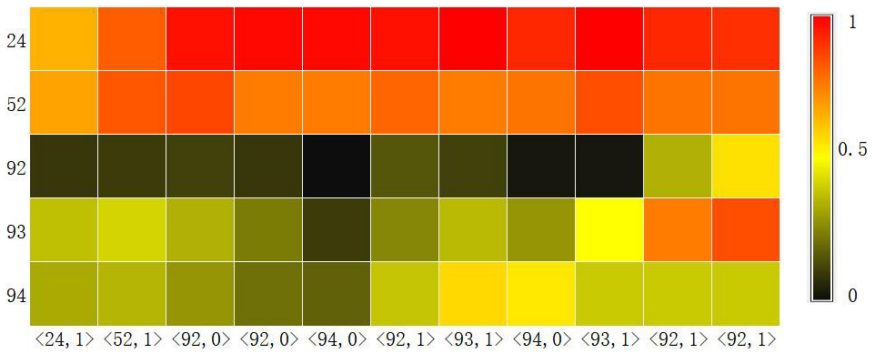

FIGURE 3 | The knowledge level output result of CAKQN-R^{nl} on the ASSISTments2009 dataset.

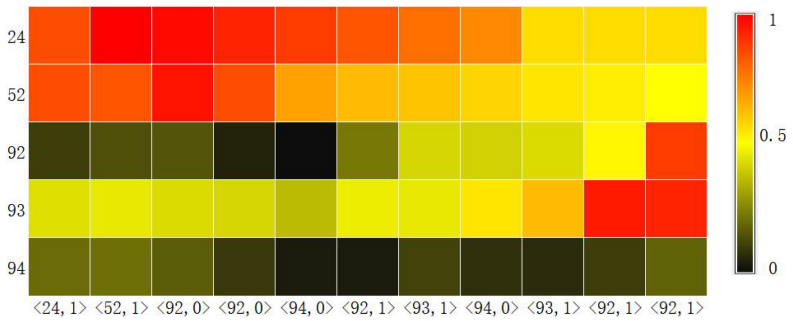

FIGURE 4 | The knowledge level output result of CAKQN-R on the ASSISTments2009 dataset.

TABLE 5 | Experimental comparison between CAKQN and CAKQN-IRT.

Model	AUC (%)	
	ASSISTments2009	**ASSISTments2017**
CAKQN-IRT	84.43 ± 0.015	75.33 ± 0.020
CAKQN-R	**87.04 ± 0.004**	**79.33 ± 0.002**

The best results are shown in bold.

predictive ability of the model. This shows that although CAKQN-IRT incorporates a more complex two-parameter IRT model, CAKQN-R has achieved better results with a simpler model structure. This also confirms that CAKQN-R has more advantages in the knowledge interaction process represented by the dot product calculation in the knowledge query component.

Visualization of Knowledge Tracking

Another basic task of knowledge tracking is to show learners' mastery of each knowledge point in real time. Therefore, we visualized the probability of learners answering correctly at each knowledge point at each time step through the Knowledge Query component. We intercepted the learning records of a learner in the dataset ASSISTments2009 over a period of time, and used the CAKQN-R^{nl} and CAKQN-R model models to track the changes in learners' mastery of 5 knowledge points, as shown in **Figures 3**, **4**. The horizontal axis in the figure represents the interception of the learner's 11 time steps of learning history. The in the tuple represents the learner's KC (knowledge points), represents the learner's answer. The vertical axis represents the 5 knowledge points tracked by the model.

From the visualization results, it can be seen that at the first time step, after the learners answered the exercises containing knowledge points 24 correctly, the tracking results of CAKQN-R^{nl} and CAKQN-R on the learners' knowledge points 24 have been improved (the probability of correct answers increases). The results indicate that the CAKQN-R^{nl} model and the CAKQN-R model will update the mastery of the corresponding knowledge points accordingly after obtaining the learner's historical answer results. In **Figures 3**, **4**, within ten time steps after the learner correctly answered the exercises containing knowledge point 24 at the first time step, CAKQN-R^{nl} did not update the learner's mastery of knowledge point 24, while CAKQN-R showed that the degree of learner's mastery of knowledge point 24 has been declining. It can be seen that the CAKQN-R^{nl} model does not consider the learner's forgetting behavior during the learning period, and the CAKQN-R model fits the learner's actual forgetting behavior during the learning period after introducing the multiplicative exponential decay term. The above results show that both the CAKQN-R model and the CAKQN-R^{nl} model can model the learning process of learners' knowledge status over time. However, the CAKQN-R^{nl} model cannot model the forgetting behavior of learners, while the CAKQN-R model can

model the forgetting behavior of learners, and more accurately track learners' mastery of various knowledge points in real time.

CONCLUSION

Real-time assessment of learners' online learning knowledge level helps to monitor learners' own cognitive status, adjust learning strategies, and improve the quality of online learning. As for four real online education datasets, this paper proposes a CAKQN model based on Rasch model embedding. It uses the vector dot product to describe the interaction process between the learner's knowledge state and the question, and uses the network structure of LSTM + monotonic attention mechanism to capture the question and the learner's personalized contextual representation. Compared with most other knowledge tracking models, it can not only track learners' knowledge status in real time, but also model learners' forgetting behavior.

However, the method presented in this paper has several limitations.

(1) CAKQN uses binary variables to represent the answer to the question as same as other KT methods. This way is not suitable for subjective questions with continuous score distribution. Wang et al. (2017) and Swamy et al. (2018). provide a new way to model subjective questions, they used continuous snapshots of the learner's answers as an indicator of the answer when dealing with learners' programming data. Modeling subjective topics will be the direction of future research.

(2) The adaptive capacity of the model needs to be improved. CAKQN is a supervised training method like other deep knowledge tracking methods, so the predictive ability of the model is dependent on the effect of training on the current dataset. If you are faced with small data sets or other domain datasets, the performance of the model may be poor (Wang Y. et al., 2021).

(3) Like most other KT methods, our method is based on the learner's historical practice record modeling, and involves too few features. In fact, the learning process is very complex, involving many other features such as the text of the question, the learning rate of the student, and the positive/negative emotions that the student generates during the learning process. At present, with the rapid development of technologies such as intelligent perception, wearable devices, and the Internet of Things, multi-modal learning analysis will become a new trend driving intelligent education research (Wang Z. et al., 2021). Under this trend, knowledge tracking will surpass a single behavior modality and gradually develop into a learner model driven by the fusion of multimodal data such as behavior, psychology, and physiology.

DATA AVAILABILITY STATEMENT

The original contributions presented in this study are included in the article/supplementary material, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

YC: writing – review and editing, supervision, resources, and conceptualization. GW: methodology, software, validation, and writing – original draft. HZ: visualization and writing – review and editing. PL: data curation. ZC: formal analysis. All authors contributed to the article and approved the submitted version.

FUNDING

This research was supported in part by the National Natural Science Foundation of China (62167006 and 61967011), Jiangxi Province Science and Technology Innovation Base Plan-Provincial Key Laboratory Project (20212BCD42001), 03 Special and 5G Projects in Jiangxi Province (20212ABC03A22), National Social Science Fund Key Project (20AXW009), Natural Science Foundation of Jiangxi Province (20202BABL202033 and 20212BAB202017), and Humanities and Social Sciences Key (Major) Project of the Education Department (JD19056). The Jiangxi Province Main Discipline Academic and Technical Leader Training Program–Leading Talent Project (20213BCJL22047).

ACKNOWLEDGMENTS

We acknowledge the financial support provided by all the fundings on this research.

REFERENCES

Bailey, C. D. (1989). Forgetting and the learning curve: a laboratory study. *Manag. Sci.* 35, 340–352. doi: 10.1287/mnsc.35.3.340

Cheng, S., Liu, Q., Chen, E., Huang, Z., Huang, Z., Chen, Y., et al. (2019). "DIRT: deep learning enhanced item response theory for cognitive diagnosis," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management (CIKM)*, (Gold Coast, AU). doi: 10.1145/3357384.3358070

Corbett, A. T., and Anderson, J. R. (1994). Knowledge tracing: modeling the acquisition of procedural knowledge. *User Model. User Adapted Interact.* 4, 253–278. doi: 10.1007/BF01099821

Daniluk, M., Rocktäschel, T., Welbl, J., and Riedel, S. (2017). Frustratingly short attention spans in neural language modeling. *arXiv [Preprint]*.

Fan, X. (1998). Item response theory and classical test theory: an empirical comparison of their item/person statistics. *Educ. Psychol. Meas.* 58, 357–381. doi: 10.1177/0013164498058003001

Ghosh, A., Heffen, N., and Lan, A. S. (2020). "Context-aware attentive knowledge tracing," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, (Virtual Event). doi: 10.1145/3394486.3403282

Hu, X., Liu, F., and Bu, C. (2020). Research progress of cognitive tracking models in educational big data. *J. Comput. Res. Dev.* 57, 2523–2546.

Khajah, M. M., Huang, Y., González-Brenes, J. P., Mozer, M. C., and Brusilovsky, P. (2014). "Integrating knowledge tracing and item response theory: a tale of two frameworks," in *Proceedings of the 4th International Workshop on Personalization Approaches in Learning Environments (PALE)*, (Aalborg).

Lee, J., and Yeung, D. Y. (2019). "Knowledge query network for knowledge tracing: how knowledge interacts with skills," in *Proceedings of the 9th International Conference on Learning Analytics & Knowledge (LAK)*, (Tempe, AZ). doi: 10.1145/3303772.3303786

Liu, T., Wei, C., Liang, C., and Gu, T. (2021). *Research Progress of Knowledge Tracking Based on Deep Learning [OL]*. Available online at: http://kns.cnki.net/kcms/detail/11.1777.TP.20210609.0938.002.html (accessed June 30, 2021).

Pandey, S., and Karypis, G. (2019). "A Self-attentive model for knowledge tracing," in *Proceedings of the 12th International Conference on Educational Data Mining (EDM)*, (Montreal, CA).

Pardos, Z. A., and Heffernan, N. T. (2011). "KT-IDEM: introducing item difficulty to the knowledge tracing model," in *Proceedings of the 19th International Conference on User Modeling, Adaptation and Personalization*, (Girona). doi: 10.1007/978-3-642-22362-4_21

Pashler, H., Cepeda, N., Lindsey, R. V., Vul, E., and Mozer, M. C. (2009). Predicting the optimal spacing of study: a multiscale context model of memory. *Adv. Neural inf. Process. Syst.* 22, 1321–1329.

Piech, C., Spencer, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., et al. (2015). "Deep knowledge tracing," in *Proceedings of the 28th International Conference on Neural Information Processing System (NeurIPS)*, (Cambridge, MA).

Rabiner, L., and Juang, B. (1986). An introduction to hidden Markov models. *IEEE ASSP Magazine* 3, 4–16. doi: 10.1109/MASSP.1986.1165342

Sun, J., Li, D., Peng, X., Zou, R., and Wang, P. (2021). Cognitive tracking from a data perspective: framework, problems and enlightenment. *Open Educ. Res.* 27, 99–109.

Swamy, V., Guo, A., Lau, S., Wu, W., Wu, M., Pardos, Z., et al. (2018). "Deep knowledge tracing for free-form student code progression," in *International Conference on Artificial Intelligence in Education*, (Cham: Springer). doi: 10.1007/978-3-319-93846-2_65

Wang, L., Sy, A., Liu, L., and Piech, C. (2017). "Deep knowledge tracing on programming exercises," in *Proceedings of the 4th ACM Conference on Learning*, (New York, NY) doi: 10.1145/3051457.3053985

Wang, Y., Wang, Y., and Zheng, Y. (2021). Multi-modal learning analysis: "Multi-modal"-driven new trends in intelligent education research. *China Audio Visual Educ.* 03, 88–96.

Wang, Z., Xiong, S., Zuo, M., Min, Q., and Ye, J. (2021). Knowledge tracking from the perspective of smart education: status quo, framework and trend. *J. Distance Educ.* 39, 45–54.

Wilson, K. H., Karklin, Y., Han, B., and Ekanadham, C. (2016). Back to the basics: bayesian extensions of IRT outperform neural networks for proficiency estimation. *arXiv [Preprint]*.

Yang, F., and Li, F. W. B. (2018). Study on student performance estimation, student progress analysis, and student potential prediction based on data mining. *Comput. Educ.* 123, 97–108. doi: 10.1016/j.compedu.2018.04.006

Yudelson, M. V., Koedinger, K. R., and Gordon, G. J. (2013). "Individualized bayesian knowledge tracing models," in *International Conference on Artificial Intelligence in Education*, (Berlin). doi: 10.1007/978-3-642-39112-5_18

Zhang, J., Shi, X., King, I., and Yeung, D. Y. (2017). "Dynamic key-value memory networks for knowledge tracing," in *Proceedings of the 26th International Conference on World Wide Web (WWW)*, (Perth, AU). doi: 10.1145/3038912.3052580

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.