Graph Knowledge Tracking Interaction Model Combining Classification and Regression Tree

Zhuoran Li[1], Tingnian He[2]*, Yixuan Rong[3], Guoqi Liu[4]

{2021222205@nwnu.edu.cn[1], hetn@nwnu.edu.cn[2],2021222260@nwnu.edu.cn[3], 2021222228@nwnu.edu.cn[4]}

Northwest Normal University, Gansu Lanzhou 730070, CHINA

Abstract. Knowledge tracking aims to assess learners' current level of knowledge based on their previous response performance. In recent years, graph neural networks have been successfully applied in the field of knowledge tracking. However, existing graph-based knowledge tracking interaction models typically combine a graph convolutional neural network with a Long Short-Term Memory network. Although there is an improvement in performance, the model only considers exercises, knowledge, and answers as inputs, ignoring the impact of rich learning behavioral features on the learner's knowledge state. In this work, we propose a Graph-based Interaction Model for Knowledge Tracing with Decision Tree (GIKT-DT) that fuses classification and regression trees. Especially, to effectively capture the effect of behavioral features on answer results, predicted responses are first obtained by pre-processing learners' behavioral features using classification and regression trees. And then the cross-sectional features of predicted responses and interaction sequences are calculated as inputs to the GIKT-DT to track learners' knowledge acquisition levels more accurately. Moreover, we validate the GIKT-DT model on three publicly available online education datasets. The experimental results show that GIKT-DT outperforms other baseline models and can better utilize the behavioral characteristics of learners to improve the accuracy and effectiveness of knowledge tracking with better prediction performance.

Keywords: Knowledge Tracing, Graph Neural Networks, Deep Learning, Decision Trees.

1 Introduction

With the development of education digitization, online education platforms such as intelligent tutoring systems are becoming widely available. Online education platforms can collect learners' interaction information in the process of practicing, and exams. By analyzing the interaction information, potential learning patterns are explored and personalized guidance is provided to learners **[1].** While knowledge tracking is the basis for online education platforms to realize personalized instruction.

Many approaches have been proposed to solve the knowledge tracking problem. Corbett et al. **[2]** proposed Bayesian Knowledge Tracking (BKT) based on Bayesian formulation. Piech et al. **[3]** proposed Deep Knowledge Tracking (DKT), which is the first deep learning knowledge tracking model that uses Recurrent Neural Networks (RNN) to model the learning process of learners. Zhang et al. **[4]** proposed Dynamic Key-Value Memory Network (DKVMN) which uses Memory Augmented Neural Network to solve the knowledge tracking problem. Pandey et

NMDME 2023, October 13-15, Xi'an, People's Republic of China

DOI 10.4108/eai.13-10-2023.2341299

al. **[5]** proposed Self-Attention Knowledge Tracking (SAKT), which uses self-attention to assign weights to previous interaction sequences. Nakagawa et al. **[6]** proposed Graph-based Knowledge Tracing (GKT), which redefines the knowledge tracing task as a time series node classification problem. Yang et al. **[7]** proposed Graph-based Interaction Model for Knowledge Tracing (GIKT). In educational psychology, the motivation theory of learning **[8]** and the attribution theory of success and failure **[9]** have studied the causes of learning behaviors and their outcomes. It is believed that the success or failure of the outcomes will be affected by internal factors such as ability, physical and mental conditions, as well as external factors such as the difficulty of the work, luck, and environment. Learners produce a large number of behavioral characteristics during the learning process, like reaction time, whether they ask for hints and the number of attempts **[10].** However, the existing graph-based knowledge tracking models do not consider behavioral characteristics.

In this paper, we propose GIKT with Decision Tree (GIKT-DT), a new graph knowledge tracking model that fuses classification and regression trees. GIKT-DT uses a classification and regression tree (CART) on top of the GIKT model for the learner's behavioral features to be preprocessed to obtain the predicted response. At the same time, Graph Neural Network (GCN) is used to process the practice-skill relationship graph, aggregating the information of neighboring nodes to obtain the higher-order information of the practice-skill, and embedding the higher-order information of the practice and the corresponding answers into the interaction sequence. After that, the cross-feature fusion method is used to obtain the cross-features of the CART decision responses and the interaction sequences, and the cross-features are used as inputs to the RNN. Finally, predictions are made using the recall module and interaction module of the GIKT model.In a nutshell, our main contributions are summarized as follows:

(1) We design a CART-based preprocessing method for behavioral features to obtain predicted responses.

(2) We propose GIKT-DT, a new graph knowledge tracking model that fuses classification and regression trees. GIKT-DT takes into account the influence of rich behavioral features on learners' response results.

(3) We conduct multifaceted validation of the model on three real online education datasets, and the experimental results show that the GIKT-DT model outperforms other baseline models.

2 Related work

2.1 CART

CART can be applied to both classification and regression problems and is considered to be the most popular decision tree algorithm today **[11]**. The CART algorithm uses a decision tree to represent the classification and regression models. Each node represents an attribute test and each edge represents a test result. CART generates a binary tree using either Gini coefficient or variance for feature selection, where the Gini coefficient measures the purity of the classification results and the variance measures the variance of the regression results. The CART algorithm has the following advantages: the learned features are interpretable, it can handle both numerical class features and categorical features, it can avoid overfitting and correctly handle missing values. Therefore, CART was chosen to preprocess the learner behavioral features in this study.

2.2 Deep knowledge tracking

The task of knowledge tracking can be defined as: extracting the learner's implicit knowledge state from the learner's interaction records $X_t = \{x_1, x_2, \cdots, x_t\}$ on the set of exercises $Q_t = \{q_1, q_2, \cdots, q_t\}$ by means of a predefined model and predicting the next answering performance presently x_{t+1}, with $x_t = \{q_t, a_t\}$ meaning that the learner answered the exercise q_t at the moment t, and a_t denoting the answering situation, usually $a_t = \{0,1\}$, with 1 denoting correctness and 0 denoting error. Existing knowledge tracking models predict the probability of answering the next question correctly usually using an alternative approach: $P(a_{t+1} = correct|q_{t+1}, X)$ **[12]**.Given the excellent performance of Recurrent Neural Networks (RNN) in handling time series tasks, researchers have tried to apply them to knowledge tracking tasks. In 2015 Piech et al **[3]** proposed Deep Knowledge Tracing (DKT), which is the first application of deep learning to the field of knowledge tracing, using RNN to model the learner's responses, taking the learner's history of responses as an input, and using the RNN hidden state vectors to represent the learner's knowledge level, to simulate the change of the learner's knowledge state in the process of learning, and to discover correlations between the exercises, and to predict the future performance of the learner based on the learner's history of response sequences.

2.3 Knowledge tracking based on graph neural networks

In recent years, research in knowledge tracking has focused on exploring the use of graph structures to capture more information. In 2020 Yang et al **[7]** proposed a graph-based interaction model for knowledge tracking. GIKT treats the practice-skill relationship as a bipartite graph and uses graph convolutional neural networks (GCNs) to obtain higher-order information between practice-skills through embedding propagation. Using long and short-term memory networks to learn learners' level of mastery of knowledge.

3 Graph knowledge tracking interaction model combining classification and regression tree

Preprocessing behavioral features so that the model takes into account the impact of behavioral features helps track the learner's mastery level of the exercise **[10]**. Therefore, we use CART to preprocess the behavioral features based on GIKT to more accurately model the learner's knowledge state and improve the predictive performance of the model. The structure of the GIKT-DT model is shown in **Figure 1**.First, a decision tree is constructed for features that may affect the learner's knowledge mastery state, and the behavioral features are preprocessed using the decision tree to predict the outcome of the answer, while the relationship between practice and skill is aggregated using a graph convolutional neural network. After that, the results of decision tree prediction and the practice results obtained from graph convolutional network are cross-featured and then used as inputs to the long and short-term memory network, through which the learners' knowledge status is obtained. Finally, the learners' responses in the future are predicted by the review module and the interaction module of GIKT.

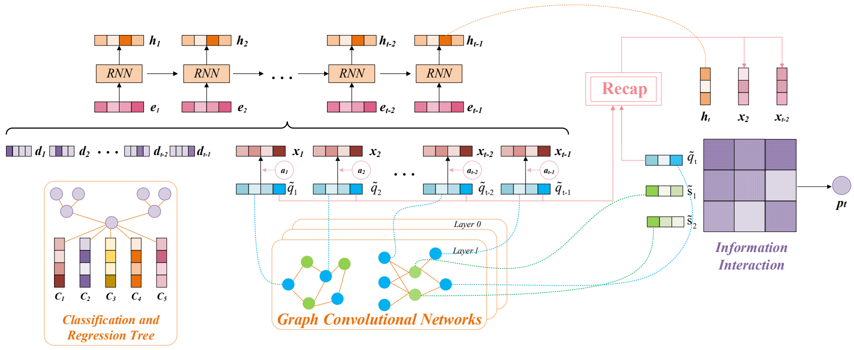

Figure. 1. Architecture for GIKT-DT

3.1 Decision Tree

We propose a CART-based preprocessing method for behavioral features that uses CART to preprocess the behavioral features that mainly affect learners' question answering, and consider the impact of behavioral states on learners' knowledge states, so as to model learners' knowledge states more accurately. In the classification task, the commonly used feature evaluation index is the Gini coefficient. In this paper, the Gini coefficient is chosen as the feature evaluation metric, and the Gini coefficient of set D is calculated by the formula:

$$Gini(D) = 1 - \sum_i p_i^2 \tag{1}$$

In Equation (1), p_i represents the probability that the sample point belongs to the i class. After that, the samples are assigned to different subnodes based on the values of the features:

$$Gini(D, C) = \sum_j \frac{|D_j|}{|D|} Gini(D_j) \tag{2}$$

Equation (2) represents the Gini coefficient of the set D after division according to feature C, and D_j represents the subset after division.The feature with the smallest Gini coefficient is selected as the partitioning standard. Finally, for each subset, the above two steps are repeated to build the subtree recursively until all the features are used.

The learner's answer results are predicted using the constructed decision tree and the answer results are encoded as d_t using solo heat coding. See Table 1 for more details.

Table 1. CART prediction response symbol interpretation

Symbol	Meaning
1000	The learner mastered the subject and answered it correctly
0100	The learner did not grasp the topic but guessed correctly
0010	The learner mastered the subject but failed in answering it
0001	The learner failed to grasp the question and gave the wrong answer

3.2 Cross-cutting features

Cross-features allow multiple features to be combined or interacted with each other to create new features to capture the nonlinear relationships between features, thus improving the model's characterization and prediction capabilities **[13]**. In this paper, we combine learner answer interaction sequences and decision tree prediction results into a new cross-feature, which enables the model to capture the interaction information between answer interaction sequences and behavioral features, which is more helpful for the model to differentiate the effect of learner's behaviors on the answer results, and to improve the prediction accuracy of the model. We use the following method to obtain the crossover features:

$$e_t = x_t + max(x+1) * d_t \tag{3}$$

where e_t represents the cross-feature, x_t is the answer interaction sequence, and d_t is the decision tree prediction result.

3.3 GIKT model

The improved GIKT consists of 5 main components:

3.3.1 Interactive Sequence Embedding

In order to solve the sparsity of interactive records, we construct a practice-skill relationship graph. Relevant exercises and skills are encoded as exercise embeddings and skill embeddings using GCN. GCN encodes higher order neighbor information by stacking multiple graph convolutional layers, in each layer the node representation can be updated by embedding itself and its neighboring nodes, the formula for the l layer GCN can be expressed as:

$$L_i^l = \sigma\left(\frac{1}{|N_i|}\sum_{j \in N_i \cup \{i\}} w^l L_j^{l-1} + b^l\right) \tag{4}$$

where L_i is denoted as node i in the graph and the set of its neighboring nodes is N_i, w^l and b^l are the aggregation weights and biases to be learned in the l GCN layer, and σ is the nonlinear transformation. Use $\tilde{q}$ and $\tilde{s}$ to denote the practice and skill representations after embedding propagation, while using the nonlinear activation function ReLU to connect the practice and answer embeddings to obtain x_t:

$$x_t = ReLU(W_1[\tilde{q}_t, a_t] + b_1) \tag{5}$$

3.3.2 LSTM

In order to obtain the learner's mastery state of knowledge, the entire practice process was modeled using LSTM and the model was made to take into account the potential relationships between the exercises.

$$i_t = \sigma(W_i[e_t, h_{t-1}, c_{t-1}] + b_i) \tag{6}$$

$$f_t = \sigma\left(W_f[e_t, h_{t-1}, c_{t-1}] + b_f\right) \tag{7}$$

$$o_t = \sigma(W_o[e_t, h_{t-1}, c_{t-1}] + b_o) \tag{8}$$

$$c_t = f_t c_{t-1} + i_t tanh(W_c[e_t, h_{t-1}] + b_c) \tag{9}$$

$$h_t = o_t tanh(c_t) \tag{10}$$

where i_t, f_t, o_t, c_t, h_t, denote the input gate, forget gate, output gate, cell state and hidden state, respectively.

3.3.3 History Recap Module

The first method is hard selection, directly considering exercises that share the same skills as the new exercise:

$$I_x = \{x_i | N_{qi} = N_{qt}, i \in [1, \dots, t-1]\} \tag{11}$$

The second method is soft selection, where the correlation between the target exercise and the historical states is learned through an attentional network and the top k states with the highest attentional scores are selected:

$$I_x = \{x_i | R_{i,t} \leq k, V_{i,t} \geq v, i \in [1, \dots, t-1]\} \tag{12}$$

where $R_{i,t}$ is the ranking of the attention function $f(q_i, q_t)$, $V_{i,t}$ is the attention value, and V is the lower bound of similarity for filtering less relevant exercises.

4 Experimentation and evaluation

In this section, we verify the performance of our model in different aspects through several experiments. First, GIKT-DT is compared with the baseline model on three real datasets. Then, it is analyzed against the decision tree designed to assess the prediction error.

4.1 Dataset

The experiments in this paper were conducted on three real and widely used datasets, which are detailed in Table 2. ASSISTments2009: This dataset contains information such as learner response records and learner behavior data from American middle school mathematics courses **[14].** The dataset has 3,823 learners, 123 skills, 17,726 exercises, and 335,748 practice data. ASSISTments2012: More learner response records and learning behavior data than 2009. The dataset has 27,485 learners, 265 skills, 53,065 exercises and 270,943,636 practice data. EdNet: Contains learning data from multiple educational platforms on the Internet **[15].** From this dataset, we randomly selected 5,000 learners, 189 skills, 12,161 exercises, and 676,974 exercises to conduct the experiment.

We used 80% of the data as a training set and 20% as a test set for each dataset. To evaluate the results on these datasets, we use the area under the curve (AUC) as an evaluation metric.

Table 2. Data set details

	ASSISTments2009	ASSISTments2012	EdNet
#students	3823	27485	5000
#questions	17726	53065	12161
#skills	123	265	189
#exercises	335748	2709436	676974
#questions per skill	172.528	200	147
#skills per question	1.197	1.000	2.280

#attempts per question	18.941	51	56
#attempts per skill	3267.401	10224	8420

4.2 Baseline model

We use the following model as our baseline model for comparison:

DKT: Proposed by Piech et al., 2015 **[3]**. RNN is used to simulate the learning process of learners, and the degree of learners' mastery of knowledge is obtained. DKT-DT: Proposed by Yang et al., 2017 **[10]**. A DKT-based tree classifier is used to predict whether the learner can answer the exercises correctly, and the predicted results are fused with the interaction sequence as input to the LSTM. DKVMN: Zhang et al proposed in 2017 **[4]**. Use static and dynamic matrices to store all skills and update the learner's state of knowledge. DKVMN-DT: Proposed by Sun et al in 2019 **[13]**. Based on DKVMN, decision tree is used to preprocess the behavior characteristics. GKT: Proposed by Hiromi Nakagawa et al., 2019 **[6]**. The knowledge tracking task is transformed into a node-level classification problem of time series. GIKT: Yang et al proposed in 2020 **[7]**. GCN is used to process the "practical skills graph" and takes the results as input to the LSTM, which is then predicted through a retrospective and interaction module. SGKT: Proposed by Wu et al in 2022 **[16]**. First, the conversation graph is used to model the students' question and answer process, and then the gated graph neural network is used to obtain the students' knowledge state from the conversation graph.

4.3 Overall Performance

The minimum batch size was set to 32. The displayed results in Table 3**.** The results of the experiment are shown in Table 3. From the results, we observe that the AUC performance of our proposed GIKT-DT on three datasets is better than that of the baseline model, which verifies the validity of our model. In more detail, our proposed GIKT-DT model is at least 1.5% higher than other baseline models. By comparing with the basic model GIKT, we find that the performance of the model is greatly improved after considering the influence of learner behavior characteristics.

Table 3. Comparison of AUC on different data sets

Model	ASSISTments2009	ASSISTments2012	EdNet
DKT	0.7376	0.7083	0.6673
DKT-DT	0.7412	0.7262	0.7189
DKVMN	0.7494	0.7274	0.7044
DKVMN-DT	0.7713	0.7506	0.7414
GKT	0.7232	0.7198	0.7122
GIKT	0.7874	0.7723	0.7502
SGKT	0.7903	0.7849	0.7519
GIKT-DT	0.8102	0.8016	0.7738

4.4 Effect of CART

On the ASSISTments2009 dataset, we use the following five behavioral features to construct the decision tree model for training us :attempt_count, which represents the number of times learners have tried an exercise; First_action, which represents the first action that the learner performs in the exercise; Hint_ total, which indicates the number of possible prompts in the exercise; Overlap_time, the amount of time it takes the learner to complete the exercise; ms_first_response, which represents the time between the start of the exercise and the learner's first action.

The trained decision tree is shown in **Figure 2.** in each node, the first row represents the split of the selected features and their thresholds. The second row "gini" represents the Gini coefficient. The third line "sample" represents the total number of samples in the node. The fourth row "value" represents the number of samples in each category. The fifth line "class" indicates whether the current node is in the right state.

The results are shown in Table 3. The AUC performance of our proposed GIKT-DT on the ASSISTments2009 dataset is 0.0318 higher than that of GIKT. The AUC values of the decision trees we trained on each dataset were 0.9970, 0.9934, and 0.9788, respectively, indicating that the decision trees we trained had high performance in making predictions based on the behavioral characteristics of learners. Decision trees have obvious advantages in interpretability. The construction of decision tree can not only find out which behavior features have an impact on learners' response results, but also compare the influence degree of different behavior features on learners by comparing root nodes and leaf nodes.

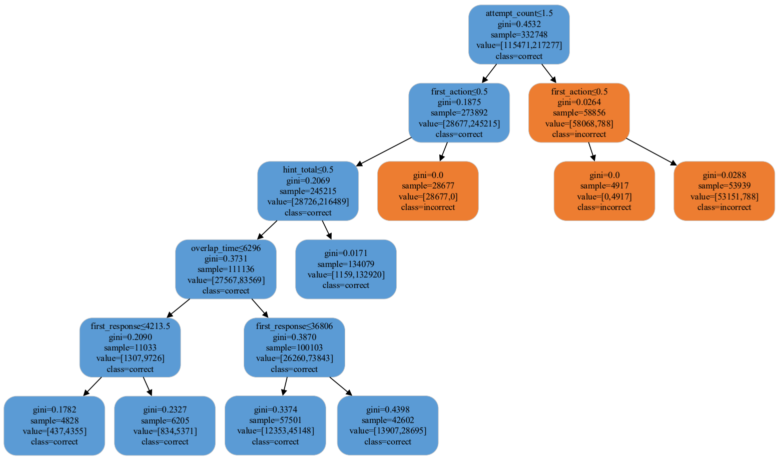

Figure 2. Decision tree that CART learns on ASSISTments09

4.5 Learner Knowledge State Evolution

One of the main tasks of knowledge tracking is to track the learner's mastery level of knowledge in real time. The visual representation of the state of knowledge can more intuitively show the strength of the learner's mastery level of the skill, and learners, teachers or researchers can more accurately carry out targeted improvement, which provides the basis for personalized teaching on online education platforms.

We intercepted a learning record of the learner with user_id 78978 from the ASSISTments2009 dataset, selected several exercises involving related skills, and constructed an exercise-skill relationship diagram as shown in **Figure 3.** The DKT, GIKT and GIKT-DT models were used to track the changes in the learners' mastery levels of these four skills respectively, and the results are shown in **Figures 4~6.** The vertical axis represents the 4 skills tracked by the model, and the horizontal axis represents the learner's history of answering questions, including the skills involved in the exercise, the answering results (0 for incorrect answers, 1 for correct answers), and the decision tree prediction results. The heat map module colors indicate the predicted probability that the learner will answer the corresponding skill correctly at the next moment. The darker the color, the higher the probability.

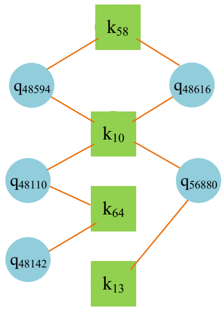

Figure 3. ASSISTment2009 part of the practice - skill relationship diagram

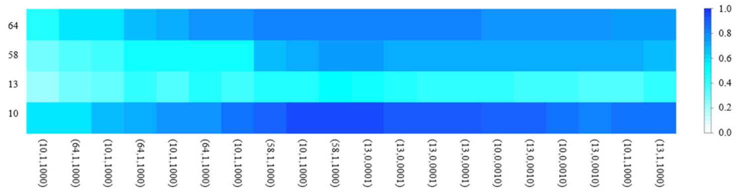

Figure 4. Cognitive state output results based on DKT

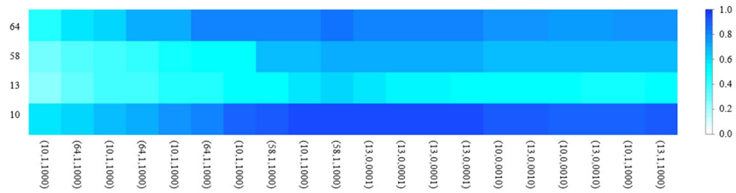

Figure 5. Cognitive state output results based on GIKT

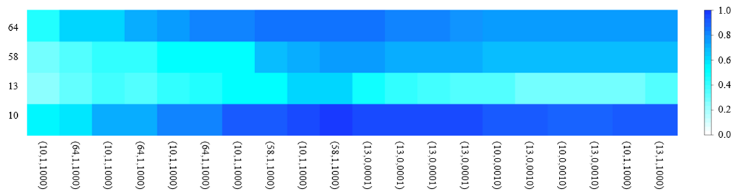

Figure 6. Cognitive state output results based on GIKT-DT

As can be seen from the experimental results:

(1) In **Figure 4.** and **Figure 6.** at moments 1, 3, 5, 7, and 9, after learners answered skill 10 correctly in succession, the results of the DKT model showed that there was no significant change in learners' mastery level of the never-answered skill 13, whereas the results of the GIKT-DT model showed that there was a slow upward trend in learners' mastery level of the never-answered skill 13. This is due to the fact that the GIKT-DT model constructs a practice-skill relationship graph and is able to do that correctly answering skill 10 affects the learners' mastery level of other skills. This proves that the GIKT-DT model is able to capture the relationships between skills between exercises, and between exercises and skills compared to the DKT model, and that the GIKT-DT is able to better track the cognitive state of learners.

(2) In **Figure 5.** and **Figure 6.** after the learners answered skill 10 and 13 incorrectly from 15 to 18, the GIKT model results showed that the learners' mastery level of skill 10 and 13 decreased more rapidly, while the GIKT-DT model results showed that the learners' mastery level of skill 10 and 13 decreased more slowly. This is due to the fact that in the decision tree prediction module of the GIKT-DT model, the prediction result for the moments from 15 to 18 is "0010", which means that the learner has mastered the skill but answered it incorrectly, and the GIKT-DT model thinks that the incorrect answering of this question is a case of error, and the influence of the learner's behavioral characteristics on the answering result is positive. This proves that the GIKT-DT model is able to capture the positive influence of learning behavioral characteristics on the answer result compared to the GIKT model, and the GIKT-DT can better track the cognitive state of learners.

The above results show that the GIKT-DT model can effectively model the relationship between exercises, between skills and between exercises and skills and the influence of learners' behavioral characteristics, and can better track learners' skill mastery level.

5 Conclusion

In this paper, we propose GIKT-DT, a new graph knowledge tracking model that fuses classification and regression trees. GIKT-DT extends the GIKT model by using a decision tree to preprocess the behavioral features to obtain a predicted response for the current exercise, providing better input information. We demonstrated the effectiveness and strengths of our approach with experiments on two publicly available datasets. In future work, we will also consider behavioral characteristics that are not directly represented in the dataset. For example, information on forgetting behavior, learner competence, etc. Using richer information about behavioral characteristics, the learner learning process is modeled.

References

[1] Abdelrahman G, Wang Q, Nunes B. Knowledge tracing: A survey[J]. ACM Computing Surveys, 2023, 55(11): 1-37.

[2] Corbett A T, Anderson J R. Knowledge tracing: Modeling the acquisition of procedural knowledge[J]. User modeling and user-adapted interaction, 1994, 4: 253-278.

[3] Lyu H, Sha N, Qin S, et al. Advances in neural information processing systems[J]. Advances in neural information processing systems, 2019, 32.
[4] Zhang J, Shi X, King I, et al. Dynamic key-value memory networks for knowledge tracing[C]//Proceedings of the 26th international conference on World Wide Web. 2017: 765-774.
[5] Pandey S, Karypis G. A self-attentive model for knowledge tracing[J]. arXiv preprint arXiv:1907.06837, 2019.
[6] Nakagawa H, Iwasawa Y, Matsuo Y. Graph-based knowledge tracing: modeling student proficiency using graph neural network[C]//IEEE/WIC/ACM International Conference on Web Intelligence. 2019: 156-163.
[7] Yang Y, Shen J, Qu Y, et al. GIKT: a graph-based interaction model for knowledge tracing[C]//Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part I. Springer International Publishing, 2021: 299-315.
[8] Dörnyei Z. Motivation in action: Towards a process-oriented conceptualisation of student motivation[J]. British journal of educational psychology, 2000, 70(4): 519-538.
[9] Zuckerman M. Attribution of success and failure revisited, or: The motivational bias is alive and well in attribution theory[J]. Journal of personality, 1979, 47(2): 245-287.
[10] Yang H, Cheung L P. Implicit heterogeneous features embedding in deep knowledge tracing[J]. Cognitive Computation, 2018, 10: 3-14.
[11] Lee D H, Kim S H, Kim K J. Multistage MR-CART: Multiresponse optimization in a multistage process using a classification and regression tree method[J]. Computers & Industrial Engineering, 2021, 159: 107513.
[12] Liu Q, Shen S, Huang Z, et al. A survey of knowledge tracing[J]. arXiv preprint arXiv:2105.15106, 2021.
[13] Wang Y, Fang Z, Hong H, et al. Flood susceptibility mapping by integrating frequency ratio and index of entropy with multilayer perceptron and classification and regression tree[J]. Journal of Environmental Management, 2021, 289: 112449.
[14] Xiong X, Zhao S, Van Inwegen E G, et al. Going deeper with deep knowledge tracing[J]. International Educational Data Mining Society, 2016.
[15] Choi Y, Lee Y, Shin D, et al. Ednet: A large-scale hierarchical dataset in education[C]//Artificial Intelligence in Education: 21st International Conference, AIED 2020, Ifrane, Morocco, July 6–10, 2020, Proceedings, Part II 21. Springer International Publishing, 2020: 69-73.
[16] Wu Z, Huang L, Huang Q, et al. SGKT: Session graph-based knowledge tracing for student performance prediction[J]. Expert Systems with Applications, 2022, 206: 117681.