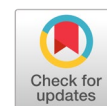


Enhanced personalized learning exercise question recommendation model based on knowledge tracing



Pei Pei ^{a,b,1,*}, Rodolfo C. Raga Jr ^{b,2}, Mideth Abisado ^{b,3}

^a Computing and Information Technologies, National University, 551 M.F. Jhocson St, Sampaloc, 1008 Metro Manila, Philippines

^b Anhui Sanlian University, No.47 He'an RD, Hefei 230061, China

¹ peip@students.national-u.edu.ph; ² rjrcraga@national-u.edu.ph; ³ mbabisado@national-u.edu.ph

* corresponding author

ARTICLE INFO

Article history

Received June 2, 2023

Revised July 1, 2023

Accepted December 18, 2023

Available online December 29, 2023

Keywords

Knowledge tracing

Personalized learning

recommendation

Graph neural network

Cognitive diagnosis

ABSTRACT

Personalized exercise question recommendation is a crucial aspect of smart education used to customize educational exercises and questions to individual students' distinct abilities and learning progress. Integrating cognitive diagnosis with deep learning has shown promising results in personalized exercise recommendations. However, the black-box nature of the deep learning model hinders their interpretability. This makes it challenging for educators and students to understand the reasons behind the model's predictions for the next problem, and this limits their opportunity to take an active role in improving the learning process. To address this limitation, this article presents a novel personalized exercise question recommendation model based on knowledge tracing. The approach incorporates graph convolutional neural networks to model the student's abilities, thus enhancing the interpretability of the model. By employing a Bidirectional gate recurrent unit (Bi-GRU), the model effectively traces fluctuations in students' abilities over time and predicts their responses to exercise questions. Experimental results demonstrate the effectiveness of this model, achieving an accuracy of 90.8% and 92.6% on ASSISTment 2009 and ASSISTment 2017 datasets, containing 4218 and 1709 student records, respectively. Moreover, the experiment was also conducted to validate the model's exercise difficulty setting. Results indicate an acceptable level of effectiveness in generating appropriate difficulty-level recommendations for individual students. The proposed model contributes to advancing personalized exercise recommendations by offering valuable insights that can lead to more efficient and effective student learning experiences.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



1. Introduction

Student's academic performance is not only a standard for certification and evaluation, but it is also essential in preparing students for their future endeavors. [1] Personalized learning, used to customize learning resources and methods to individual learners' distinct abilities and learning progress, has emerged as a promising approach to complement online learning and achieve better development. Usually, researchers consider the issue from the educational psychology perspective and propose cognitive diagnosis models to discover students' knowledge proficiency [2]. Cognitive diagnosis is widely applied in the field of education. However, most models are linear. They are with limitation to learn the complex interaction between students and exercise questions. At present, considerable progress in personalized learning has been achieved through methods based on cognitive psychology and related theories, however, personalized learning in online scenarios still faces the issue of information overload [3]. In

addition, when provided with numerous exercise resources, students find it difficult to select suitable exercises to their abilities. To address these challenges, the development of a personalized exercise recommendation model has drawn inspiration from e-commerce recommendation systems. The model leverages students' history of answering records to build individual ability models and predict future performance, ultimately recommending exercises at an appropriate ability level.

In response to the above problems, we will review relevant literature in the fields of cognitive diagnosis and knowledge tracing. The work [4] provides a HELP-DKT model embedded students ability by Q -matrix achieved better prediction. The work in [5] proposed a novel model incorporated three embedding: students, exercises and skills to reach a better prediction performance. It aims to reduce the impact of the subjective labeling by calibrating the skill relation matrix and the Q -matrix and updates the heterogeneous interactions between students, exercises, and skills by graph convolutional network. The work in [6] pointed out that existing research are still lack of certain performance due to their neglect of complete content-based exercising, fine-grained knowledge concepts, and cognitive labels for specific requirement. Meanwhile, cognitive diagnosis is frequently used as psychological measurement in educational psychology area. It is considered as an evaluation method of students' mastery level for various knowledge concepts. The DINA model (Deterministic Inputs Noise and gate model) proposed by Torre [7] is a classic cognitive diagnosis model in the field of education. This model uses the incidence Q matrix of knowledge points to enhance the interpretability of cognitive diagnosis results, Qi Bin [8], Shan Ruiting [9] and other researchers all use the characteristics of Q matrix to ensure the accuracy and interpretability of recommendation results in combination with collaborative filtering. In response to the problem of insufficient predictive ability of the DINA model, some researchers have proposed an improved model for DINA. Researchers such as Latore [10] and Tu Dongbo [11] proposed a multi-level diagnostic structure with higher precision based on the problem of DINA scoring only 0-1. Researchers such as Liu [12] and Li Youxi [13] proposed a fuzzy cognitive diagnostic method that considers the importance of knowledge points. Jiang Peichao [14] constructed students' knowledge state by quantifying their potential cognitive level towards learning materials, and proposed a cognitive diagnostic method that combines students' reading materials. He Xiangnan [15] and other researchers proposed a general framework of neural collaborative filtering (NCF), which introduces neural networks to learn the interaction information between users and goods, and its prediction effect is better than linear methods such as matrix decomposition. Wang [16] and other researchers proposed the NeuralCD method for education based on the NCF method. This framework combines neural networks to learn the complex interaction between students and test question vectors and uses the monotonicity hypothesis and Q matrix of educational psychology for reference to ensure the interpretability of students' ability model. DKT-LCIRT [17] emphasized on reflect intrinsic difference between students by kinds of capability vectoring, therefore the model can articulately present interpretability.

Knowledge tracing [18] was first proposed in 1994 and has become a hot research topic in the field of smart education in recent years with the development of RNN [19]. The task of knowledge tracing can be described as: given a student's record of problem-solving, tracking the student's level of mastery of knowledge points and predicting the situation of the next test question. Chris Piech [20] and other researchers proposed deep knowledge tracing, applying RNN to the field of knowledge tracing, using a recurrent neural network model to process input student sequences, tracking the dynamic changes in students' abilities over time, and ultimately predicting students' problem-solving performance. The predictive ability of DKT significantly exceeds that of BKT, but due to the lack of introduction of relationships between knowledge points and exercise questions, the model has interpretable issues. Yeung [21] and other researchers proposed the Deep IRTM, which combines the item response theory with the dynamic key value memory network model and uses IRT to simulate the state transformation.

This paper makes notable contributions to this field of study in two key areas:

- By proposing a cognitive diagnostic model based on graph convolutional networks, which involves constructing a student ability model and modeling the interaction process of student exercises using nonlinearity. The utilization of the Q matrix ensures the interpretability of the diagnostic results.

- By introducing a knowledge tracing model grounded in the personalized recommendation to accurately predict students' exercise performance and generate a list of recommended exercises based on their difficulty levels. Experimental verification demonstrates the effectiveness of setting the recommended difficulty level.

2. Method

To solve the challengeable issues mentioned above, we propose a personalized exercise recommendation model based on knowledge tracing (PERKT). The graph convolutional neural network algorithm is implemented to construct an aggregated student ability vector, which improves the performance of the model while using educational domain related limitations to train the model to ensure the interpretability of the student ability model. By utilizing of graph convolutional neural network to construct a student ability model with deep interaction features, ensuring the interpretability of the student ability model. Taking Gate Recurrent Unit (GRU) to track the dynamic changes of student ability models over time, and generate a recommended list of test questions based on predicted student performance and difficulty. Moreover, deep learning is used to trace the changes in student ability with order and time characteristics, Finally, generate a recommendation list based on the predicted student response and difficulty range of the exercise questions.

2.1. Architecture

PERKT architecture was proposed in Fig.1. The process began with reading the annotated input data as students' log from online learning (the left box on top Fig.1). The Q matrix was provided to represent the correlation between test questions and knowledge topics.

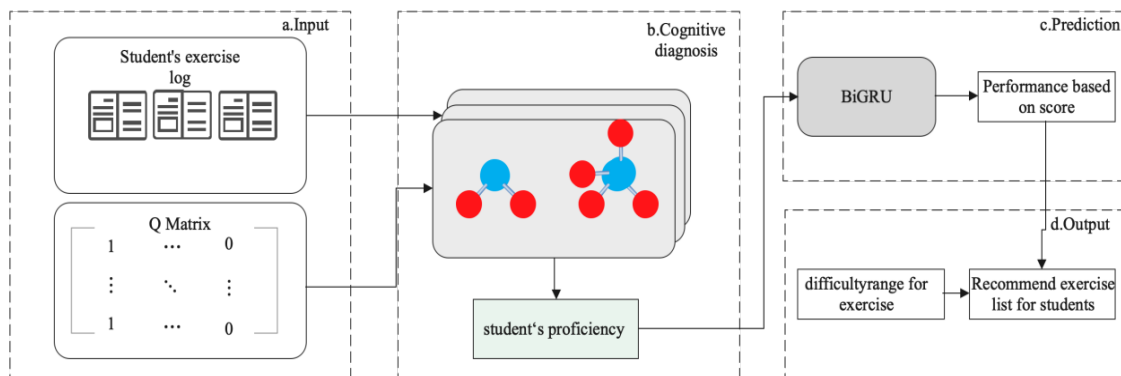


Fig. 1. Framework of PERKT (authors enhanced model)

2.2. Cognitive Diagnosis Based on Graph Convolutional Neural network

GCN (Graph Collaborative Network) [22] has become the latest technology of collaborative filtering since it was first proposed by ICLR2017. There are two basic nodes in GCN: user and item. Based on their associations, a user item bipartite graph is constructed, and the representation of these nodes is learned by smoothing features in the graph [23]. The execution graph convolution of GCN iteration, where the new representation of the target node is aggregated from the features of its neighbors. Cognitive diagnosis makes use of the diagnosis vector to model the interaction process between students and exercises [16]. In this study, we consider four aspects, including knowledge point correlation vector Q_e , student ability vector a_s , student vector v_s and exercise vector v_e .

- Knowledge point correlation vector Q_e

This vector represents the relationship between exercise and knowledge concepts. It will combine with student ability vector as well as exercise differentiation degree as the input of neural network for prediction. It will improve the interpretability of cognitive diagnosis based on the student's level related to each knowledge point. We can get Q_e from the product of exercise one-hot coding and Knowledge related Q matrix.

$$Q_e = x^e \times Q \quad (1)$$

- Student vector v_s

Students vector v_s can be calculated through student one-hot coding multiplied by a parameter matrix A.

$$v_s = \text{sigmoid}(x^s \times A) \quad (2)$$

- Exercise vector v_e

Exercise vector v_e is aim for assisting the computation based on graph convolution neural network of the students vector. It has the same dimension of v_s , by vector polymerization.

$$v_e = f(v_s) \quad (3)$$

- Student ability vector a_s

Both student vector and exercise vector are aim for modeling student ability vector. Student ability vector is a quantitative representation of students' proficiency in knowledge points and concepts. In this paper, students' proficiency in each knowledge point concept is expressed in a continuous way, and this more refined way represents students' proficiency in knowledge points, which is conducive to improving the accuracy of grade prediction in the interaction process of modeling students' exercise questions, and also helps students to conduct self-assessment according to the cognitive diagnosis results.

$$a_s = \{a_{s1}, a_{s2}, \dots, a_{sD}\} \quad (4)$$

Where, the dimension of a_s is the same as Q_e , and $a_{sD} \in [0,1]$ represents student's mastery of knowledge point D. This combination of students' ability vector and knowledge quantity correlation vector makes the model have good interpretation, and because of the aggregation of neighbor information in the process of graph convolution, the model has more accurate diagnosis results.

- Student ability modeling

The traditional cognitive diagnostic methods use directly interacting exercise questions with students to model their abilities, ignoring the common problem-solving characteristics of similar students. The collaborative filtering method extracts the commonness of similar students by calculating the similarity of students, but it ignores the influence of high concatenate of similar students' exercise answering [8]. For the above issues, this article proposes a graph convolution algorithm for modeling students' abilities, which can capture the feature of exercising from higher concatenate of neighbor node and more accurately construct students' ability models. In the process of student answering exercise, multiple students who have had direct interaction with the same test question usually have similar exercise-solving characteristics, which can be used to discover commonalities among students. The indirect high concatenate between students and exercise questions is difficult to explore through direct interaction, and using high concatenate connectivity can more accurately explore the commonalities between students. This algorithm takes students and exercise questions as the basic nodes in the graph convolutional network for multi-layer convolution calculations, so that each student and question node in the graph convolution aggregates information from nodes with excessive interaction. This concatenate aggregation can be abstracted as:

$$v_s^{(l+1)} = g(v_s^{(l)}, \{v_e^{(l)}: e \in \mathcal{N}_s\}) \quad (4)$$

In this algorithm, students and exercises are taken as the basic nodes in the graph convolutional neural network for multi-layer convolution calculation. Make each student and exercise node in the convolution of the graph aggregate nodes of high-order interaction, which can be expressed as:

Where, g is the aggregate function used to calculate the normalized sum of the directly interactive student exercises. In this algorithm, the student ability vector is represented by the fusion of the nodes of the convolution of each layer of graph, and the student ability vector with the aggregation of multi-layer cooperative information can reflect the student ability more accurately. The student-exercise interaction is regarded as a dynamic evolution process, and the problems done by similar students can be regarded as the characteristics of common students. Through the graph convolution operation, the characteristics of higher-order neighbors doing problems can be encoded into the students' ability vector. This process can be viewed as a kind of collaborative filtering.

Fig. 2 shows the calculation process of ability vector of student s_2 . From top to the bottoming the first layer, by aggregating students who have done the same exercise questions, a representation of the test question node is formed in the second layer. This test question node contains the common problem-solving features of these students, and the nodes in each subsequent layer are calculated in the same way. The number of layers in graph convolution represents the information of students or exercise questions that a student can aggregate as far as possible. After multi-layer graph convolution operation, the student representations of each layer are fused, and the problem-solving features of high-order neighbor students in the graph can be encoded into the student's ability vector. Neighbors on each layer can be considered as similar students (or exercises), so aggregating higher-order neighbor features can be seen as a collaborative filtering.

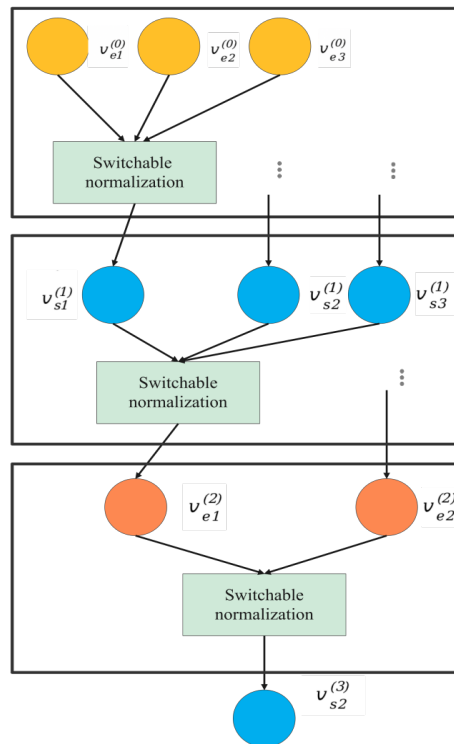


Fig. 2. Modeling the Process of student s_2 proficiency vector

We can express the graph convolution calculation as:

$$v_s^{(l+1)} = \sum_{e \in N_s} \frac{1}{\sqrt{|N_s|} \sqrt{|N_e|}} v_e^{(l)} \tag{5}$$

$$v_e^{(l+1)} = \sum_{s \in N_e} \frac{1}{\sqrt{|N_e|} \sqrt{|N_s|}} v_s^{(l)} \tag{6}$$

Among them, the denominator term is the symmetric normalization term, which can avoid the size of student vector and exercise vector increasing with the convolution calculation of graph. l represents the number of layers of graph convolution, and $l=0$ represents the initial student vector and item vector. In order to avoid overfitting, the three-layer graph convolution network is used in this model to ensure the performance of the model [24]. In addition, linear calculation method is adopted for each layer node to reduce the difficulty of training. After three layers of convolution, the student vector obtained from each layer is fused as the student ability vector:

$$a_s = \sum_{l=0}^3 p_l v_s^{(l)} \quad (7)$$

where, p is the parameter of each layer in the graph convolutional network:

2.3. Prediction based on knowledge tracing

The goal of KT is to estimate knowledge mastery of students based on their historical answering performance of related exercises. But the current deep learning model are with limitation in: (a) they focus on the details of the nodes rather than to high-level interactive information; (b) they struggle to effectively establish complex structures of the nodes; and (c) they represent either students or exercises only, without integrating them [25]. In order to accurately predict students' performance, we propose a bidirectional GRU model to tracing students' historical ability levels. The model is shown in the figure below (Fig. 3).

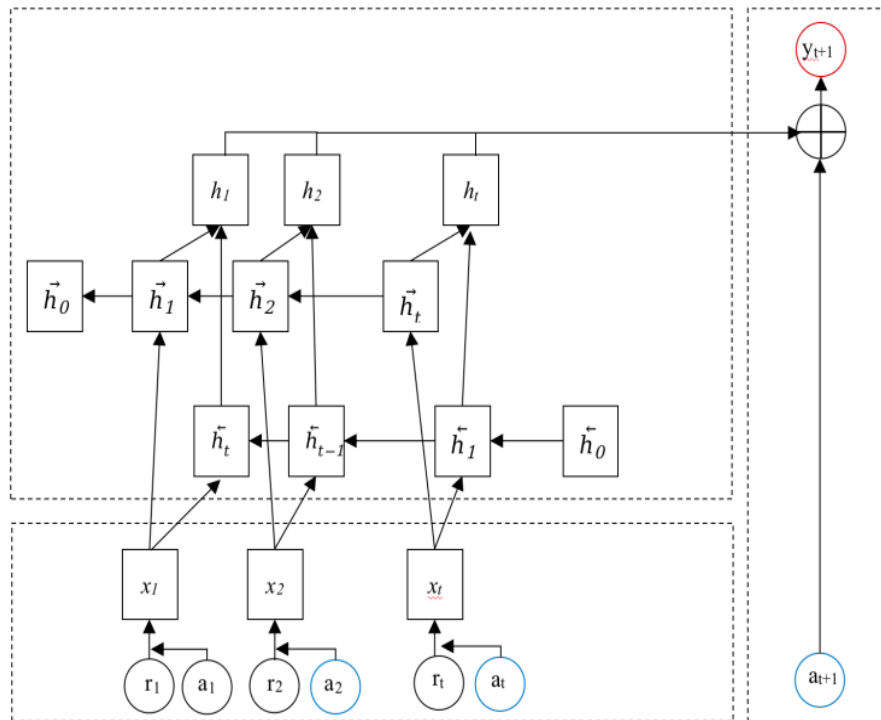


Fig. 3. BiGRU model

For this model, the input is the interaction sequence $X_t = \{x_1, x_2, \dots, x_t\}$ of students' exercise questions, where the student's ability level a_t obtained from the graph convolution cognitive diagnosis process and the corresponding answer r_{tto} the test question form a historical test question interaction tuple $x_t = (a_t, r_t)$ for students, representing their problem-solving situation with a level of a_t , mastery of knowledge points. By learning the characteristics of students' historical ability level sequence through bidirectional GRU, students' problem-solving behavior can be effectively modeled, and the hidden layer h and the ability level of students $a_{(t+1)}$ at time $t + 1$ can be fused, ultimately predicting students' answering situation at time $t + 1$.

In the process of modeling students' problem-solving behavior, specific student ability levels are used as inputs, with h representing the number of forward hidden units $\vec{h}_t \in \mathbb{R}^{n \times h}$ time step reverse hidden state and use $\vec{h}_t \in \mathbb{R}^{n \times h}$ as forward hidden state. At the same time, two GRU neural networks in opposite directions are trained, and their hidden layers are connected to the same output, The information of these two directional states is simultaneously obtained by the output layer, and the formula for calculating the hidden vector can be defined as:

$$\begin{aligned}
 z_t &= \sigma(W_z \cdot [h_{t-1}, x_t]) \\
 p_t &= \sigma(W_r \cdot [h_{t-1}, x_t]) \\
 \tilde{h} &= \tanh(W_{\tilde{h}} \cdot [p_t * h_{t-1}, x_t]) \\
 h_t &= (1 - z_t) * h_{t-1} + z_t * h_t \\
 y_t &= \sigma\sigma(W_0 \cdot h_t) \\
 h_t &= \vec{h}_t \oplus \tilde{h}_t
 \end{aligned} \tag{8}$$

Where in the formula $W_0, W_z, W_{\tilde{h}}$ and W_r respectively the corresponding trainable weight matrices, x_t representing the ability level tuple at the current time, $h_{(t-1)}$ representing the output of the network at the previous time, z_t representing the update gate, p_t representing the reset gate, $[\]$ and $*$ respectively, represent the connection of the matrix and the multiplication of matrix elements, $y_{(t+1)}$ representing the probability of students correctly answering the exercise questions at $t + 1$ time.

In the process of model optimization, it is necessary to calculate the difference between the real value and the predicted value and minimize the loss function. The parameters that need to be updated in this model include two parts: parameter matrices $\{A, B\}$ in cognitive diagnosis and parameter matrices $\{W_0, W_z, W_{\tilde{h}}, W_r\}$ Knowledge tracing prediction process. The model is constructed by using the binary cross entropy loss function, whose formula is:

$$L = -\sum_{t=1}^T (r_t \log y_t + (1 - r_t) \log(1 - r_t)) \tag{9}$$

Among them, T represents the size of the input sequence, r_t representing the actual student performance at time step t and the y_t representing predicted performance. This model uses the Adam algorithm [26] to optimize the model, which can more effectively update the parameter values of the network compared to the cascade descent algorithm.

2.4. Recommendation of exercise questions based on difficulty range

The final step of this model is to provide personalized recommend exercise questions for different students based on the predicted student responses mentioned above. In the context of education, the recommended exercise questions for students should be those with difficulty levels that match their abilities, and this difficulty level should be within a specific range according to teaching needs. For this purpose, we adopted a recommendation method that can set a difficulty range. The difficulty of the exercise reflects the difficulty scale of the test question for students. The difficulty D_{es} of the test question is denoted as: exercise e for students s is set to the probability that students s can correctly answer the exercise e , which is defined as:

$$D_{es} = P(r_{es} = 1 | e, s) \tag{10}$$

In the process of recommending exercise questions, in order to clarify the difficulty boundary of the recommended exercise questions, the difficulty range β_1 and β_2 are set, $\beta_1 < \beta_2$. Among them, we can predict the student's answer situation y and recommend the set of exercise questions from $[\beta_1, \beta_2]$ within the set of exercise questions E , where the predicted numerical range can be defined as:

$$E_s = \{E_e | E_e \in E, D_{es} \in [\beta_1, \beta_2]\} \quad (11)$$

Among them, β_1 and β_2 respectively, represent the upper and lower bounds on the probability of students correctly answering questions. After setting the difficulty range for the test question set, a personalized recommendation list of difficult questions can be generated for different students. For example, to give students a certain challenge, a difficulty range parameter of $\beta_1 = 0.1$ to $\beta_2 = 0.2$ is designed, and then the model will select the exercise questions with a correct answer probability of 0.1 to 0.2 in the test question list for recommendation, that is, for $y \in [0.1, 0.2]$.

3. Results and Discussion

3.1. Experimental Setup

Datasets and preprocessing. This experiment was conducted using the ASSISTments2009 [27] and ASSISTments2017 datasets. And we choose the previous proposed work [28] that exclude data of duplications. Table 1 summarizes basic statistics of the datasets.

The experiment was conducted using the skill builder sub data from the ASSISTments2009 dataset. This includes 4218 students, 346860 student test question interactions, 17726 exercise questions, and 123 knowledge points, with an average of 1.20 knowledge points per question. Among them, id represents the question number of the test question, 0 and 1 represent the student's answer status. While the ASSISTments2017 dataset included 1709 students, 3162 exercise questions, 102 knowledge points, and 942816 student question interactions, with an average of 1.94 knowledge points per question.

Table 1. Description of Datasets

Dataset	ASSISTment2009	ASSISTment2017
Students	4218	1709
Exercises	17726	3162
Knowledge concepts	123	102
Number of interaction	346860	942816
Average number of knowledge concept per exercise item	1.20	1.94

In the implementation, we use the environment of the experiment is macOS BigSur operating system, the processor uses Apple M1, and the memory is 8G. The python language and PyTorch framework are used to build the model. We divide the data into two datasets for students' problem-solving situations. 80% of the dataset is used for the training set and the rest for the testing set. The experiment adopts five-fold cross verification.

We reviewed several state-of-the-art modeling to compare including IRT, BKT, PMF and DKT:

- The IRT [29] model models students' problem-solving process through logistic functions, and is a cognitive diagnostic model.
- The BKT [30] model is a classic model proposed in 1990s that assumes that each student's knowledge state is a set of binary variables, and utilizes hidden Markov models to track these variables separately. [31], [32] made some progress in this area recent years.
- The PMF [33] model is a factorization model that maps students and test questions to potential vectors.
- The DKT [17] model is a RNN or LSTM based neural networks to model the interactive process of student question answering for prediction.

We use accuracy, precision, and the ROC curve as performance indicators. Among them, ACC is the percentage of correct predictions among all results, and a higher ACC value indicates a strong expression ability of the model. The accuracy is the percentage of correctly predicted positive samples in the actual

positive samples. The AUC value ranges from 0 to 1, and for a random guess, its AUC value is 0.5. A higher AUC value indicates a higher predictive performance of the model.

3.2. Difficulty range parameter experiment

This experiment was designed to verify the validity of the parameter setting for the recommendation results. In order to verify whether the recommended difficulty exercises fall within the difficulty range set by the model parameters. In this experiment, the correct response rate index SR was adopted to evaluate the true difficulty of students in answering the recommended exercises, which represented the correct response probability of students in the exercise set E_S with the set difficulty range parameters β_1 and β_2 , and was defined as:

$$SR = \frac{\text{correct}}{\text{total}} \quad (11)$$

Where, total represents the number of exercises in the recommended exercises set, correct represents the number of correct answers of students in the set, and SR indicates the real difficulty of the recommended exercises for students. When the difficulty range is set too low, the recommended exercise list generated by the model will bring some challenges to students. When the difficulty range is set too high, the recommended exercises are easier for students to answer, and the true correct rate of students should be high.

In order to verify this model, the DKT model is used as the comparison model. In this experiment, the difficulty range of exercises was divided into 11 intervals according to the interval of 0.1. For each difficulty parameter, $[\beta - 0.1, \beta + 0.1]$ was taken as the difficulty range of exercises for experiment. The experimental results are shown in Fig. 4.

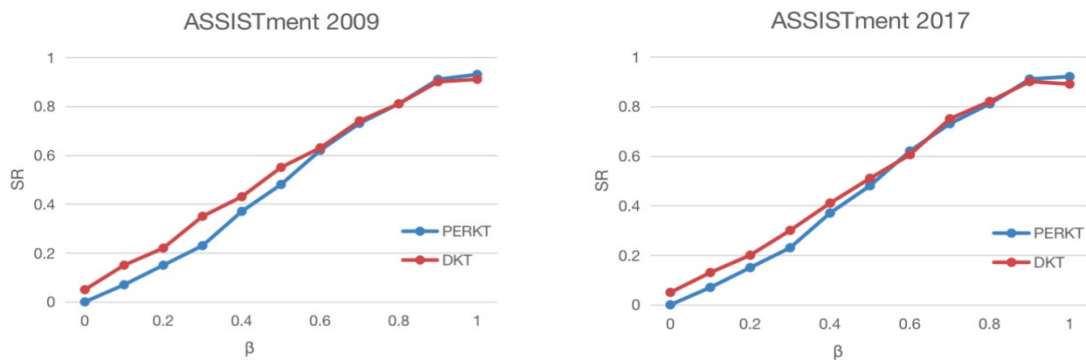


Fig. 4. SR for different difficulty ranges

As can be seen from the figure, the correct response rate SR of students on the recommended exercises continuously increases with the decrease of the difficulty range parameters, indicating that PERKT can effectively recommend exercises that meet the difficulty requirements for students. When set to 0.6, PERKT has an SR value close to 0.6 on both data sets. However, due to the lack of more detailed modeling for students' mastery of knowledge points, the degree of individualization of DKT is not enough, and its SR values are 0.63 and 0.66 respectively. Moreover, on the data set with more interactive data of students' exercises, the SR values of PERKT are more stable. The experimental results show that PERKT uses graph convolutional neural network to build a more accurate student ability model, which can recommend exercises of different difficulty to students personalized and has better interpretability.

3.3. Interpretability of predictions

In order to further verify the interpretability performance of the PERKT model, consistency experiments were conducted in this section. The degree of correlation between the predicted academic scores of consistency assessment and the knowledge points. This section compares the consistency of BKT, DKT and PERKT on the ASSISTments2009 and ASSISTments2017 datasets, and sets random

results for reference. As shown in Fig. 5, since PERKT uses Q matrix and monotonicity assumption to update parameters when building student ability model with graph convolutional neural network, its DOA (Degree of Agreement) value is higher than other comparison models. The experiment shows that the prediction results of PERKT have good interpretability.

It is generally believed that if students x have a higher level of mastery of knowledge point concepts than students y , it indicates that students x have a higher probability of correctly answering exam questions containing knowledge points than students y in the field of education [34]. The form of the consistency of knowledge points $DOA(k_c)$ is:

$$DOA(k_c) = \frac{1}{Z} \sum_{x=1}^N \sum_{y=1}^N \delta(\alpha_{xk}, \alpha_{yk}) \sum_{e=1}^M \frac{Q_{ek} \wedge J(e, x, y) \wedge \delta(r_{xk}, r_{yk})}{Q_{ek} \wedge J(e, x, y)} \quad (12)$$

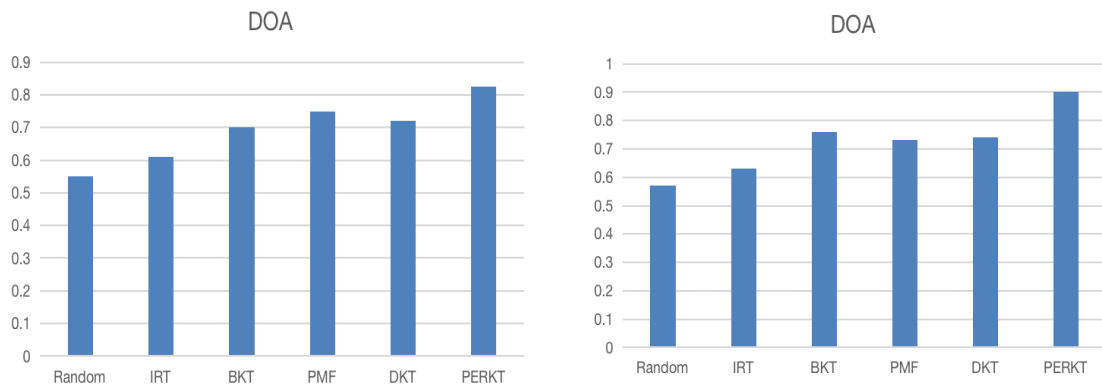


Fig. 5. DOA of different models

Among these indicators, the students x have higher proficiency in knowledge points than students y . this is represented as a non-zero value, otherwise it is 0. Equation $Z = \sum_{x=1}^N \sum_{y=1}^N \delta(\alpha_{xk}, \alpha_{yk})$ indicates student x master more knowledge points than student y . $J(e, x, y) = 1$ represent student x did exercise e but y did not, otherwise is 0. Finally, taking the mean of all consistency as the evaluation indicator $DOA \in [0, 1]$, $DOA = 0.5$ indicates that the cognitive diagnostic results of the model are not correlated with the predicted results of student performance, and the higher the DOA value show the stronger correlation between the two.

3.4. Experimental Results and Discussion

Table 2 presents the comparison results between the PERKT model and other comparative models on relevant performance indicators.

Table 2. The prediction of students answering questions performance of different models from 5 trials

Model	Performance					
	Acc		Precision		Auc	
	ASSIST09	ASSIT17	ASSIST09	ASSIT17	ASSIST09	ASSIT17
IRT	0.757	0.781	0.739	0.835	0.694	0.718
BKT	0.638	0.695	0.681	0.673	0.708	0.723
PMF	0.762	0.794	0.737	0.756	0.714	0.751
DKT	0.832	0.857	0.850	0.912	0.868	0.833
PERKT	0.913	0.919	0.887	0.919	0.908	0.926

The experimental results show that the GCKT model outperforms other comparative models in terms of ACC, precision, and AUC on the ASSISTments2009 and ASSISTments2017 datasets. Comparing the experimental results, it can be found that the BKT model performs poorly in both datasets, indicating that the hidden Markov model used by the BKT model has limited ability to model

knowledge points and cannot capture complex student problem-solving interactions. The IRT and PMF models did not utilize the time series characteristics of students' answering, resulting in poor performance. The DKT model utilizes recurrent neural networks to trace students' knowledge levels and has good predictive performance. However, due to the direct use of student coding as input, it does not learn concatenate features between students, exercise, and knowledge points, resulting in its predictive performance being inferior to PERKT. From the experimental results, our model performs better than other comparative models in predicting students' response performance.

To demonstrate the exercise recommendation results from different model, we also select two baseline model related to student-exercise collaborative filter topics to compare. Which are:

- Student-based collaborative filter [35] (SB-CF): This model refers to the idea of collaborative filtering to find interesting content for specific users, recommend exercises based on students' similarity, build a similarity matrix between students according to students' doing records, and then identify the top 10 students whose answers are most similar to those of target students. Then select suitable difficulty questions from the answer records of each similar student for recommendation.
- Exercise-base collaborative filter [36] (EB-CF): The model refers to the idea of similarity between items to recommend, using the intrinsic quality or inherent attributes of the project to recommend, in the exercise recommendation according to the students' exercise answers to set the difficulty weight for each exercise, and then calculate the exercise similarity matrix and extract exercises similar to the exercises done, and then recommend according to the desired exercise difficulty weight.

The accuracy of exercise recommendation performance show in Table. 3.

Table 3. The Accuracy of exercise recommendation performance

Accuracy	ASSIST2009		ASSIST2017	
	Mean	Std.D	Mean	Std.D
SB-CF	0.492	0.212	0.688	0.210
EB-CF	0.653	0.101	0.391	0.213
PERKT(with modeling student's proficiency and difficulty range)	0.895	0.086	0.863	0.075

4. Conclusion

In this paper, we proposed an enhanced personalized learning exercise question recommendation model based on knowledge tracing. We combined Q matrix to improve the accuracy of modeling students' abilities and interpretability for diagnosis. Also, we borrowed the success of Bi-GRU to learn the characteristics of students' historical ability level sequences related and model their answering behavior. Finally, LSTM is used to trace the dynamic changes in students' abilities over time and predict their performance. We also implement an experiment of DOA to validate the rationality of cognitive diagnosis, and design a difficulty range parameter experiment to verify the effectiveness of the recommended exercises. Experimental results on two datasets showed the achievements of our model. In the future work, we are interested in extending attention mechanisms to improve our model performance. The psychological characteristic is also worth exploring, such as a student may make mistakes in exercises due to nervousness during the answering process.

Acknowledgment

This work was partly supported by the Key Science Planning Project of Anhui Province (Grant No.2022AH051997) and partly by the school key Science Program of Anhui Sanlian University.

Declarations

Author contribution. Pei Pei, Rodolfo C. Raga Jr. and Mideth Abisado conceived of the presented idea. Pei Pei developed the theory and performed the computations, verified the analytical methods. Rodolfo C. Raga Jr. and Mideth Abisado encouraged Pei Pei to investigate exercise recommendation for personalized learning and supervised the findings of this work. All authors discussed the results and contributed to the final manuscript.

Funding statement. This work was partly supported by the Key Science Planning Project of Anhui Province (Grant No.2022AH051997) and partly by the school key Science Program of Anhui Sanlian University.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper.

Data and Software Availability Statements

The datasets involved in this paper are available in these two links:

- ASSIT2009: <https://sites.google.com/site/assistmentsdata/home/assistent-2009-2010-data>.
- ASSITments2017:
https://drive.google.com/drive/folders/0B8VPP_XVUJoaGhvOV9rVERrY0E?resourcekey=0-L4-rsmCvBht-Sp_kQzjbiQ&usp=sharing

References

- [1] M. V. Amazona and A. A. Hernandez, "Modelling Student Performance Using Data Mining Techniques," in *Proceedings of the 2019 5th International Conference on Computing and Data Engineering*, May 2019, pp. 36–40, doi: [10.1145/3330530.3330544](https://doi.org/10.1145/3330530.3330544).
- [2] M. Liu, P. Shao, and K. Zhang, "Graph-Based Exercise- and Knowledge-Aware Learning Network for Student Performance Prediction," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13069 LNAI, Springer Science and Business Media Deutschland GmbH, 2021, pp. 27–38, doi: [10.1007/978-3-030-93046-2_3](https://doi.org/10.1007/978-3-030-93046-2_3).
- [3] T. He, "Information Overload and Interaction Overload as Two Separate Attributes of Information Overload Syndrome," *J. Enterprising Cult.*, vol. 28, no. 03, pp. 263–279, Sep. 2020, doi: [10.1142/S0218495821500114](https://doi.org/10.1142/S0218495821500114).
- [4] Y. Liang, T. Peng, Y. Pu, and W. Wu, "HELP-DKT: an interpretable cognitive model of how students learn programming based on deep knowledge tracing," *Sci. Rep.*, vol. 12, no. 1, p. 4012, Mar. 2022, doi: [10.1038/s41598-022-07956-0](https://doi.org/10.1038/s41598-022-07956-0).
- [5] L. Li and Z. Wang, "Knowledge relation rank enhanced heterogeneous learning interaction modeling for neural graph forgetting knowledge tracing," *PLoS One*, vol. 18, no. 12, p. e0295808, Dec. 2023, doi: [10.1371/journal.pone.0295808](https://doi.org/10.1371/journal.pone.0295808).
- [6] J. Yu *et al.*, "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*, Jul. 2023, pp. 2924–2934, doi: [10.1145/3539618.3591898](https://doi.org/10.1145/3539618.3591898).
- [7] J. de la Torre, "DINA Model and Parameter Estimation: A Didactic," *J. Educ. Behav. Stat.*, vol. 34, no. 1, pp. 115–130, Mar. 2009, doi: [10.3102/1076998607309474](https://doi.org/10.3102/1076998607309474).
- [8] Q. Bin, Z. Hong-xia, W. Yu, and L. Ji-xing, "Questions Recommendation Based on Collaborative Filtering and Cognitive Diagnosis," *Computer Science*, vol. 14, no. 11, pp. 235–240, 2019. [Online]. Available at: <https://www.jsjx.com/EN/10.11896/jsjx.180901827>.
- [9] S. Rui-Ting, L. Yi-Cheng, and S. Yi, "Collaborative Filtering Algorithm Based on Cognitive Diagnosis," *Comput. Syst. Appl.*, vol. 27, no. 3, pp. 136–142, 2018. [Online]. Available at: http://www.c-s-a.org.cn/csaen/article/abstract/6239?st=article_issue.
- [10] J. de la Torre and J. A. Douglas, "Higher-order latent trait models for cognitive diagnosis," *Psychometrika*, vol. 69, no. 3, pp. 333–353, Sep. 2004, doi: [10.1007/BF02295640](https://doi.org/10.1007/BF02295640).

- [11] T. Dong-Bo, C. Yan, D. Hai-Qi, and D. Shu-Liang, "A Polytomous Cognitive Diagnosis Model: P- DINA Model," *Acta Psychol. Sin.*, vol. 42, no. 11, pp. 1011–1020, 2011, doi: [10.3724/SP.J.1041.2010.01011](https://doi.org/10.3724/SP.J.1041.2010.01011).
- [12] Q. Liu *et al.*, "Fuzzy Cognitive Diagnosis for Modelling Examinee Performance," *ACM Trans. Intell. Syst. Technol.*, vol. 9, no. 4, pp. 1–26, Jul. 2018, doi: [10.1145/3168361](https://doi.org/10.1145/3168361).
- [13] Y. Li, Y. Wen, X. Yi, and Z. Xu, "Revised Model of Fuzzy Cognitive Diagnosis Framework," *J. data Acquis. Process.*, vol. 32, no. 5, pp. 958–969, Apr. 2018. [Online]. Available at: <http://sjcj.nuaa.edu.cn/sjcjcl/article/abstract/S201705013>.
- [14] J. Peichao, W. Chuan, H. Fuzhen, L. Qi, and W. Xiaodong, "Student Performance Prediction Based on Reading Cognitive Diagnosis," *Comput. Eng. Appl.*, vol. 58, no. 11, p. 160, Jun. 2022. [Online]. Available at: <http://cea.ceaj.org/EN/10.3778/j.issn.1002-8331.2010-0276>.
- [15] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural Collaborative Filtering," in *Proceedings of the 26th International Conference on World Wide Web*, Apr. 2017, pp. 173–182, doi: [10.1145/3038912.3052569](https://doi.org/10.1145/3038912.3052569).
- [16] F. Wang *et al.*, "Neural Cognitive Diagnosis for Intelligent Education Systems," *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 04, pp. 6153–6161, Apr. 2020, doi: [10.1609/aaai.v34i04.6080](https://doi.org/10.1609/aaai.v34i04.6080).
- [17] G. Li *et al.*, "DKT-LCIRT: A Deep Knowledge Tracking Model Integrating Learning Capability and Item Response Theory," *Electronics*, vol. 11, no. 20, p. 3364, Oct. 2022, doi: [10.3390/electronics11203364](https://doi.org/10.3390/electronics11203364).
- [18] A. T. Corbett and J. R. Anderson, "Knowledge tracing: Modeling the acquisition of procedural knowledge," *User Model. User-Adapted Interact.*, vol. 4, no. 4, pp. 253–278, Dec. 1995, doi: [10.1007/BF01099821](https://doi.org/10.1007/BF01099821).
- [19] R. J. Williams and D. Zipser, "A Learning Algorithm for Continually Running Fully Recurrent Neural Networks," *Neural Comput.*, vol. 1, no. 2, pp. 270–280, Jun. 1989, doi: [10.1162/neco.1989.1.2.270](https://doi.org/10.1162/neco.1989.1.2.270).
- [20] C. Piech *et al.*, "Deep Knowledge Tracing," in *Advances in Neural Information Processing Systems*, 2015, vol. 28, pp. 505–513. [Online]. Available at: <https://stanford.edu/~cpiech/bio/papers/deepKnowledgeTracing.pdf>.
- [21] C.-K. Yeung and H. Kong, "Deep-IRT: Make Deep Learning Based Knowledge Tracing Explainable Using Item Response Theory," *EDM 2019 - Proc. 12th Int. Conf. Educ. Data Min.*, pp. 683–686, Apr. 2019. [Online]. Available at: <https://arxiv.org/abs/1904.11738>.
- [22] T. N. Kipf and M. Welling, "Semi-Supervised Classification with Graph Convolutional Networks," *5th Int. Conf. Learn. Represent. ICLR 2017 - Conf. Track Proc.*, pp. 1–14, Sep. 2016. [Online]. Available at: <https://arxiv.org/abs/1609.02907v4>.
- [23] F. Wu, T. Zhang, A. Holanda De Souza, C. Fifty, T. Yu, and K. Q. Weinberger, "Simplifying Graph Convolutional Networks," in *Proceedings of Machine Learning Research*, 2019, p. 11. [Online]. Available at: <https://proceedings.mlr.press/v97/wu19e/wu19e.pdf>.
- [24] S. Peng and T. Mine, "A Robust Hierarchical Graph Convolutional Network Model for Collaborative Filtering," *arXiv*, p. 11, Apr. 2020. [Online]. Available at: <https://arxiv.org/abs/2004.14734>.
- [25] X. Song, J. Li, Q. Lei, W. Zhao, Y. Chen, and A. Mian, "Bi-CLKT: Bi-Graph Contrastive Learning based Knowledge Tracing," *Knowledge-Based Syst.*, vol. 241, p. 108274, Apr. 2022, doi: [10.1016/j.knosys.2022.108274](https://doi.org/10.1016/j.knosys.2022.108274).
- [26] D. P. Kingma and J. L. Ba, "Adam: A Method for Stochastic Optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR, pp. 1–15, Dec. 22, 2014. [Online]. Available at: <https://arxiv.org/abs/1412.6980v9>.
- [27] X. Xiong, S. Zhao, E. G. Van Inwegen, and J. E. Beck, "Going Deeper with Deep Knowledge Tracing," in *International Educational Data Mining Society*, 2016, pp. 545–550. [Online]. Available: <https://eric.ed.gov/?id=ED592679>.
- [28] X. Xiong, S. Zhao, E. G. Van Inwegen, and J. E. Beck, "Going Deeper with Deep Knowledge Tracing." 2015, [Online]. Available at: <https://github.com/chrispiech/DeepKnowledgeTracing>.

- [29] L. Peng and C. Wei, "Online Item Response Theory (OIRT) - Tracking Student Abilities in Online Learning System," *Proc. 15th Int. Conf. Educ. Data Mining, EDM 2022*, p. 720, 2022. [Online]. Available at: <https://zenodo.org/records/6853004>.
- [30] Z. Machardy and Z. A. Pardos, "Evaluating The Relevance of Educational Videos using BKT and Big Data," *Proc. 8th Int. Conf. Educ. Data Min.*, pp. 424–427, 2015, [Online]. Available at: <https://files.eric.ed.gov/fulltext/ED560875.pdf>.
- [31] S. Xu, M. Sun, W. Fang, K. Chen, H. Luo, and P. X. W. Zou, "A Bayesian-based knowledge tracing model for improving safety training outcomes in construction: An adaptive learning framework," *Dev. Built Environ.*, vol. 13, p. 100111, Mar. 2023, doi: [10.1016/j.dibe.2022.100111](https://doi.org/10.1016/j.dibe.2022.100111).
- [32] L. Meng, M. Zhang, W. Zhang, and Y. Chu, "CS-BKT: introducing item relationship to the Bayesian knowledge tracing model," *Interact. Learn. Environ.*, vol. 29, no. 8, pp. 1393–1403, Nov. 2021, doi: [10.1080/10494820.2019.1629600](https://doi.org/10.1080/10494820.2019.1629600).
- [33] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in *Advances in Neural Information Processing Systems 20 - Proceedings of the 2007 Conference*, 2008, pp. 1–8, [Online]. Available at: https://proceedings.neurips.cc/paper_files/paper/2007/file/d7322ed717dedf1eb4e6e52a37ea7bcd-Paper.pdf.
- [34] Y. Chen *et al.*, "Tracking Knowledge Proficiency of Students with Educational Priors," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, Nov. 2017, vol. Part F1318, pp. 989–998, doi: [10.1145/3132847.3132929](https://doi.org/10.1145/3132847.3132929).
- [35] a Toscher and M. Jahrer, "Collaborative filtering applied to educational data mining," *KDD Cup*, pp. 1–11, 2010, [Online]. Available at: http://pslclatashop.org/KDDCup/workshop/papers/KDDCup2010_Toeschler_Jahrer.pdf.
- [36] Z. Wu, M. Li, Y. Tang, and Q. Liang, "Exercise recommendation based on knowledge concept prediction," *Knowledge-Based Syst.*, vol. 210, p. 106481, Dec. 2020, doi: [10.1016/j.knosys.2020.106481](https://doi.org/10.1016/j.knosys.2020.106481).