IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 2, 2020, accepted October 18, 2020, date of publication October 22, 2020, date of current version November 6, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3033200

Multiple Features Fusion Attention Mechanism Enhanced Deep Knowledge Tracing for Student Performance Prediction

DONG LIU[1,2], YUNPING ZHANG[1], JUN ZHANG[3], QINPENG LI[1], CONGPIN ZHANG[1,2], AND YU YIN[4,5]

[1]School of Computer and Information Engineering, Henan Normal University, Xinxiang 453007, China
[2]Big Data Engineering Laboratory for Teaching Resources and Assessment of Education Quality, Xinxiang 453007, China
[3]School of Mechanical Engineering, Zhengzhou University, Zhengzhou 450001, China
[4]School of Computer Science and Technology, University of Science and Technology of China, Hefei 230052, China
[5]School of Data Science, University of Science and Technology of China, Hefei 230052, China

Corresponding author: Dong Liu (liudonghtu@gmail.com)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB1701402, and in part by the National Natural Science Foundation of China under Grant 62072160.

ABSTRACT Student performance prediction is a fundamental task in online learning systems, which aims to provide students with access to active learning. Generally, student performance prediction is achieved by tracing the evolution of each student's knowledge states via a series of learning activities. Every learning activity record has two types of feature data: student behavior and exercise features. However, most methods use features that are related to exercises, such as correctness and concepts, while other student behavior features are usually ignored. The few studies that have focused on student behavior features through subjective manual selection argue that different student behavior features can be used in an equivalent manner to predict student performance. In this paper, we assume that the integration of student behavior features and exercise features is crucial to improve the precision of prediction, and each feature has a different impact on student performance. Therefore, this paper proposes a novel framework for student performance prediction by making full use of both student behavior features and exercise features and combining the attention mechanism with the knowledge tracing model. Specifically, we first exploit machine learning to capture feature representation automatically. Then, a fusion attention mechanism based on recurrent neural network architecture is used for student performance prediction. Extensive experiments on a real-world dataset show the effectiveness and practicability of our approach. The accuracy of our method is up to 98%, which is superior to previous methods.

INDEX TERMS Student performance prediction, knowledge tracing, recurrent neural network, attention mechanism.

I. INTRODUCTION

With the growth of massive Internet-based educational resources, many online learning platforms have emerged, such as ASSISTments, Khan Academy and Massive Open Online Courses (MOOCs). These platforms provide students with open learning resources and help students learn in an active manner, which can optimize their knowledge structure. Student performance prediction is considered to be an important task in the development of online learning platforms and forecast whether students will answer future questions correctly [1]. Student performance prediction aims to identify students' knowledge proficiency or learning ability of students. Specifically, an exact method can reduce students' learning burdens by offering personalized learning programs and saveing time and energy for teachers by providing immediate feedback [2].

Online learning platforms generate large-scale educational data based on students' learning trajectories, which consist mainly of student behavior data and exercise data. Student behavior features describe the background information of students and summarize their clickstream records (e.g., numbers of time the student practices and whether students use hints), which is generated whenever a student answers a

The associate editor coordinating the review of this manuscript and approving it for publication was Utku Kose.

question in the online learning system. Exercise features describe the related knowledge concepts and the results of students' response. The availability of massive data resources can improve the accuracy of predicting student performance to a large extent [3]. However, using various features data for performance prediction is challenging.

In recent years, there have been a series of research achievements in the field of predicting student performance, mainly including cognitive diagnosis [4], knowledge tracing [5] and deep learning [6]. Deep knowledge tracing (DKT) [6] is gaining more attention in this area, and it is the first time that deep learning can be combined with knowledge tracing. DKT has had great success in processing the sequential education data, mainly utilizing the recurrent neural network (RNN) [7] to trace students' knowledge states. Although DKT achieves more impressive performance for knowledge tracing tasks than other models, there is some space for improvement in the accuracy of the DKT model.

The DKT model focuses on the usage of exercise results and ignores the impact of students' behavior features, which may have a certain influence on students' performance. The current work considers the impact of the exercise features or only a few behavior features on student performance, ignoring the impact of other student behavior features such as opportunity (the number of times the student practices a skill). It is worth mentioning that Zhang *et al.* proposed the extension of the DKT model, which first explored the inclusion of a few features (such as students' response time, attempt numbers and first actions) to improve its accuracy [8]. Despite their achievements, there are limitations in this work regarding the manual selection of features from subjective perceptions. Current methods select features manually and assign the same weights to each feature. Furthermore, student performance is correlated with a long-term historical learning state [9]. For RNN, the last hidden state of the coding layer alone cannot contain all features, and there is a risk that more important information will be lost.

In response to the above issues, this paper proposes a novel multiple features fusion attention mechanism enhanced deep knowledge tracing (MFA-DKT) framework that makes full use of both student behavior features and exercise features. Specifically, first, all features generated by the online education platform are incorporated into the knowledge tracing model. Second, to comprehensively analyze the impact of each feature and the relationship between features on student performance, a machine learning model is employed to automatically handle multiple features and learn their representation. This can solve the problems existing in multiple feature variables, such as noise, within the range of information loss. Third, a recurrent neural network (RNN) is used to trace the knowledge states of students by combining their sequential exercise results with feature representations. Finally, in the student performance prediction stage, combining the attention mechanism to assign different weights to features and retain more important information over time. In this way, the MFA-DKT framework can naturally predict student performance based on their learning activity records. The experimental results show that the features of the well-designed fusion Attention Mechanism in our framework have better performance.

The rest of the paper is organized as follows. The second part describes the related work. The third part introduces the problem definition and details of our framework. Our experimental dataset and results are presented in the fourth part. Finally, the fifth part summarizes our work and discusses further work to be addressed in the future.

II. RELATED WORK

At present, student performance prediction approaches can be classified into three main categories: cognitive diagnosis, knowledge tracing and deep learning. Cognitive diagnosis is a combination of cognitive psychology and educational measurement. Knowledge tracing traces the students' knowledge mastery levels over time. The last category is deep learning, which uses neural networks to predict student performance, such as recurrent neural networks and memory augmented neural networks (MANN) [10].

A. COGNITIVE DIAGNOSIS

Cognitive diagnosis is used in educational psychology to discover students' knowledge states through their exercise records [4]. Item response theory (IRT) is a typical cognitive diagnosis model that offers several parameters (such as students' latent traits, discrimination and difficulty of exercises) and uses the logistic-like IRT formula to predict student performance [11]. There is another typical cognitive diagnosis model called deterministic inputs, noisy-and gate model (DINA), which is combined with the Q-matrix (the relation of exercises and knowledge) prior to judging students' degrees of mastery in each knowledge concept [12].

B. KNOWLEDGE TRACING

Bayesian knowledge tracing (BKT) [5] is the most classical method used to trace students' knowledge states. BKT is mainly based on students' exercise records to compute four key parameters for estimating the probability that students will answer exercises correctly. However, the implementation of BKT has only skill-specific parameters. In order to consider the variability of students, Yudelson proposed individualized bayesian knowledge tracing model that incorporates student-specific parameters (such as the initial knowledge mastery level and the learning speed of students) [13].

C. DEEP LEARNING

It has become increasingly common to use deep learning methods to solve problems in many domains. Many educational researchers combine deep learning with cognitive diagnosis and knowledge tracing in the area of student performance prediction.

1) COMBINE COGNITIVE DIAGNOSIS WITH DEEP LEARNING

Traditional IRT requires specific parameters for the problem of diagnosis, and it ignores rich information in question texts. Thus, enhancing item response theory for cognitive diagnosis (DIRT) enhances the process of diagnosing parameters by exploring the exercises' text using deep learning techniques [14]. In order to capture the complex relation of students and exercises, neural cognitive diagnosis for intelligent education systems (neural CD) proposes the incorporation of neural networks [15].

2) COMBINE KNOWLEDGE TRACING WITH DEEP LEARNING

Chris Piech proposes the deep knowledge tracing (DKT) model, which is the first to apply deep learning to predict student performance, which greatly improves the accuracy predictions. It takes the time sequence into account by leveraging a variant of recurrent neural network long short-term memory (LSTM). Zhang *et al.* proposes a model called incorporating rich features into deep knowledge tracing that use a few features based on DKT. Subsequently, to determine the relationship between exercises and knowledge concepts, dynamic key-value memory networks for knowledge tracing (DKVMN) model is proposed. DKVMN designs two memory matrixes, including a static key matrix (which stores the knowledge concepts) and a dynamic value matrix (which updates students' knowledge mastery level) based on memory-augmented neural networks [16]. Prerequisite-driven deep knowledge tracing (PDKT-C) incorporates the knowledge structure information and uses another type of recurrent neural network called gated recurrent unit (GRU) [17]. At the same time, it also solves the issue of sparse data [1]. Then, exercise-enhanced sequential modeling for student performance prediction (EERNN) makes use of students' exercise records and the text of exercises to model students' states. It designs a bidirectional LSTM to capture the representation of exercises from the description of text and leverages RNN for student performance prediction [18].

D. MOTIVATION

Some of the differences between our work and previous methods are presented. First, this paper takes advantage of all the student features generated on the platform. Second, a machine learning method is applied to learn feature representation in this paper. Finally, our framework uses the attention mechanism to trace longer sequences of exercises, which is useful for capturing students' true learning states.

III. METHOD

In this section, this paper defines the notations and problems related to student performance prediction. Then, the details of the RNN and attention mechanism are described. Finally, an overview of our framework is provided in this part.

A. PROBLEM DEFINITION

The goal of predicting student performance is typically to determine whether students can correctly answer the next exercise [2]. In our work, student performance prediction makes full use of students' learning activity records, including student behavior features and exercise features.

We denote U for a set of students and E for a set of exercises. We further denote students' exercise results as $M \in \{0, 1\}$ matrix, in which 1 indicates that the student answer the exercise correctly and 0 indicates that the student answer the exercise incorrectly. At time t, the feature vector of student i is denoted as $s_{(i,t)} = \{f_1, f_2, \cdots, f_n\}$ (f_p represents the p-th feature of student and n is the length of characteristic sequence). And the feature matrix denotes $S_i = [s_{(i,1)}, s_{(i,2)}, \cdots, s_{(i,t)}]^T$(the historical learning sequence of student i from time 1 to t). We mark the results of the student $i \in U$ does exercise $j \in E$ as $r_{ij} \in \{0, 1\}$, where $r_{ij} = 1$ if the student i answers correctly and $r_{ij} = 0$ otherwise.

Definition 1 (Student Performance Prediction): According to the historical learning activity records $S_i = [s_{(i,1)}, s_{(i,2)}, \cdots, s_{(i,t)}]^T$ of each student from exercising time 1 to t, our goal is to predict the response score r_{ij} on the next exercise at time $t + 1$.

For easier checking, a list of notations mentioned in this paper is summarized in Table 1.

TABLE 1. A list of notations shown in this paper.

Notations	Description
U	a set of students shown in the data
E	a set of exercises appeared in the data
M	exercising results of students
f_k	one of learning features of student i
$s_{(i,t)}$	the learning features of student i at time t
S_i	the historical learning sequence of student i from time 1 to t
x_i	a $\|K\|$-dimensional vector after PCA method process $s_{(i,t)}$
X_i	the exercising records of student i from time 1 to t
r_{ij}	binary value of whether student i answer exercise j correctly
r'_t	the actual score of student i at time t

B. RECURRENT NEURAL NETWORK

A recurrent neural networks (RNN) is a class of artificial neural networks that is powerful for modeling sequence data. In contrast to traditional neural networks, RNN can achieve sustained memory understanding and allow information retention. The goal of RNN is to map an input sequence $\{x_1, x_2, \cdots, x_t\}$to an output sequence $\{y_1, y_2, \cdots, y_t\}$ [19]. As show in Figure 1, disregarding the attention layer is the structure of the RNN. In the process from input to output, the input vector experiences several conversions through a hidden layer, which gains available information and produces a sequence of hidden states. However, RNN suffer from gradient explosion and insensitivity to long-term information.

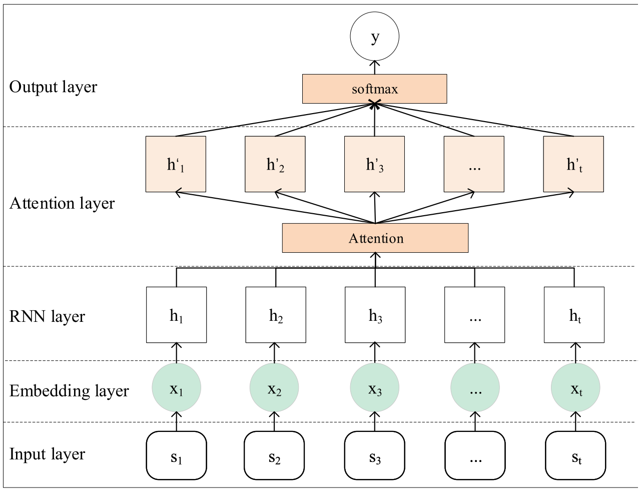


FIGURE 1. The architecture of Attention based on RNN. Input is the primary features and output is the results of prediction.

To solve this problem, a variant of RNN called long short-term memory network (LSTM) [18] has been proposed. LSTM shows unique performance in many fields, especially for processing longer sequences of data. A common LSTM unit is consist of a cell, an input gate, an output gate and a forget gate. The cell remembers values over arbitrary time intervals and the three gates regulate the flow of information into and out of the cell. LSTM can capture long-term information and remove or add dispensable/indispensable information to the cell state over time.

The goal of knowledge tracing is based on students' past learning status. Students learn through gradual process, so when tracing students' knowledge states, we must consider the effect of time series on the results. In addition, students' learning levels are constantly updated because they learn the corresponding knowledge concepts within a certain period of time or forget them [20]. Considering that a student's status is not only related to time series, but also to changes according to the students' learning levels, we chose LSTM to model student sequence.

C. ATTENTION MECHANISM

The attention model [21] has become an important concept in neural networks and has been extensively studied in different application areas such as speech recognition [22] and image annotation [23]. It is based on a common-sensical intuition that we focus on a certain segment when processing a large amount of information. Intuitively, the attention mechanism is a measure of the vector of importance weights, or a representation of the correlation between elements. When a sequence is too long, it is difficult for the hidden layer state of LSTM include long input information, and some important information may be lost. The cells of LSTM share the same weight. In fact, the student learning process is a longer learning sequence and different features have different effects on student performance. Therefore, attention mechanisms is based on LSTM to better predict student performance.

The attention mechanism is implemented by retaining the intermediate output of the LSTM encoder on the input sequence. A model is then trained to learn selectively from input and to correlate the output sequence with it at model output. Figure 1 shows the architecture of attention based on RNN. A detailed description of each layer of the structure is as follows:

Input layer: The feature sequence of the students.

Embedding layer: The students' feature embedding sequence $X_i = \{x_1, x_2, \cdots, x_t\}$ as input into the RNN model to update the hidden state.

RNN layer: Encoding the knowledge state of students according to the current input and previous hidden state.

Attention layer: A new state $h_t^{'}$ is a weighted sum aggregation of all historical student states during the process.

Output layer: After encoding the final knowledge state of students by using the softmax function to capture the result of prediction.

D. FRAMEWORK

To enhance deep knowledge tracing for student performance prediction, this paper proposes a novel MFA-DKT framework. As shown in Figure 2, MFA-DKT contains four modules: input, deep learning, attention and prediction modules. The input module processes the learning activity records of each student into a feature vector using a machine learning model. Deep learning module mainly employs deep learning method to model students' knowledge states with their learning behavior features. The attention module assigns different weights to input features to extract critical and important information. Next, in the prediction stage, MFA-DKT can forecast each student's performance in next exercise. The specific implementation of the proposed framework is shown in Figure 2.

1) FEATURE EMBEDDING

It is necessary to incorporate some influential and effective features related to students to predict their performance. Multiple features can reflect rich information; however, they also increase complexity and present problems. Thus, in the case of minimal information loss, it is necessary to reduce the feature space on the basis of fully considering feature independence and the relationship between features. Principal component analysis (PCA) [24] is an appropriate way to reduce the analysis targets and minimize the loss of information. Therefore, the PCA method is used to handle the multiple features of students in this paper. PCA can be interpreted as dimensionality reduction from the initial space to the encoded space, which is also called latent space. The details of feature embedding are described below.

The details of representing multiple features as input vectors using the PCA method to model students' learning sequences are following. First, to remove the unit restriction of multiple features and convert it into a dimensionless, pure numerical value, each sequential feature $s_{(i,j)} = \{f_1, f_2, \cdots, f_n\}$ is normalized.

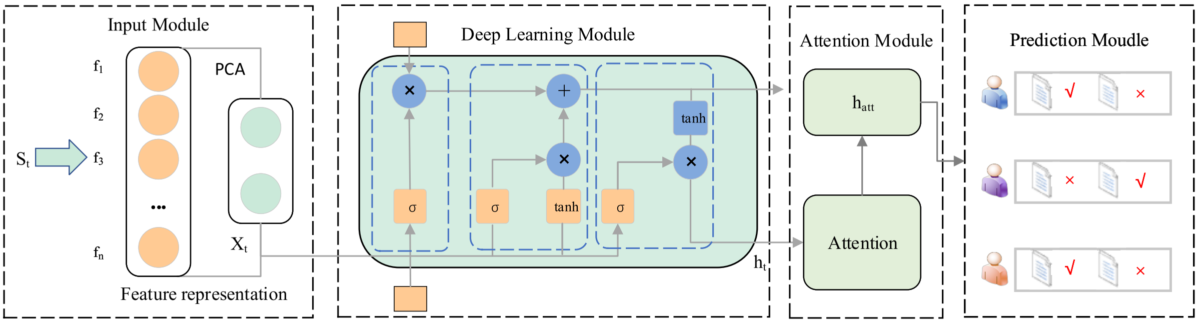


FIGURE 2. An overview of our framework.

We convert each feature sequence $\{f_1, f_2, \cdots, f_n\}$ as:

$$f_i' = \frac{f_i - \overline{f}}{m}, \overline{f} = \frac{1}{n}\sum_{i=1}^{n} f_i, m = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(f_i - \overline{f})^2} \quad (1)$$

where $\overline{f}$ is mean, m represents standard deviation and f_i' represents new feature after normalization. We get a new sequence $s'_{(i,t)} = \{f_1', \cdots, f_n'\}$ and feature matrix:

$$S_i' = [s'_{(i,1)}, \cdots, s'_{(i,t)}]^T \in R^{t\times n} \quad (2)$$

Second, we calculate the covariance matrix D:

$$D = \frac{1}{t} S_i' S_i'^T \in R^{n\times n} \quad (3)$$

The eigenvectors and their corresponding eigenvalues are then calculated through D. We sort the eigenvalues from largest to smallest. By doing so, the corresponding eigenvectors are sorted. This method selects the first k largest eigenvalues and corresponding eigenvectors to form a new matrix $P \in R^{n\times k}$. Finally, the final matrix $X_i = S_i'P, S_i \in R^{t\times k}$ is calculated. We exploit the PCA method to reduce the dimension of features and capture a matrix $X_i = \{x_1, x_2, \cdots, x_t\}$ denoting the exercising records of students i from time 1 to t, where $x_i = Dp_i, i = \{1, 2, \cdots, t\}, l = \{1, 2, \cdots, k\}$, and p_l denotes the l-th principal eigenvector of D.

It is worth mentioning that we use PCA to process multiple features automatically instead of manual selection.

2) MODELING STUDENT SEQUENCE

After obtaining each feature representation from feature embedding, $\{x_1, x_2, \cdots, x_t\}$ is input into the long short-term memory network (LSTM) for training, providing predictions of the students' response y_{ij} at $t+1$, and the specific formulae are as follows:

$$\begin{aligned}
i_t &= \sigma(w_i x_t + u_i h_{t-1} + b_i)\\
c_t &= \tan(w_c x_t + u_c h_{t-1} + bc)\\
f_t &= \sigma(w_f x_t + u_f h_{t-1} + b_f)\\
c_t &= i_t \widetilde{c_t} + f_t c_{t-1}\\
o_t &= \sigma(w_o x_t + u_o h_{t-1} + v_o c_t + b_o)\\
h_t &= o_t \tanh(c_t)
\end{aligned} \quad (4)$$

where i_t, f_t, o_t represent input, forget, output gate respectively. $\widetilde{c_t}$ represents the cell state when input feature vector passes through input gate at time t, and c_t represents the cell state combine information from both input and forget gate. $w_i, w_c, w_f, w_o, u_i, u_c, u_f, u_o, v_o$ denote weight coefficients and b_i, b_c, b_f, b_o denote bias. They are all parameters of model. $\sigma(x)$ and $\tan(x)$ are non-linear activation function. After all the parameters initialization, the model computes the hidden state of each students.

At step $t + 1$, the student state is a weighted sum aggregation of all historical student states during the process. Formally, in the next step $t + 1$, the attentive student state vector h_{att} is defined as:

$$h_{att} = \sum_{j=1}^{t} \alpha_j h_j, \alpha_j = softmax(h_j) \quad (5)$$

where h_j is hidden state at j and softmax is activation function. α_j is attention score for measuring the importance of features. After obtaining attentive student state h_{att} at step $t + 1$, combining the current input x_{t+1} to output the response of students r_{t+1}. The specific formulae are as follows:

$$\begin{aligned}
y_{t+1} &= \sigma(w_h x_{t+1} + w_l h_{att} + b_l)\\
r_{t+1} &= softmax(y_{t+1})
\end{aligned} \quad (6)$$

where y_{t+1} denotes overall presentation for prediction at $t+1$ exercise step. $\{W_h, W_l, b_l\}$ are the parameters.

3) OBJECTIVE FUNCTION

In order to make a comparison of the real label, it is necessary to transform the real label into a two-dimensional vector by using one-hot encoding. The list [0, 1] denotes the answer is correct and [1, 0] denotes the answer is incorrect in the corresponding exercise. The element on the left side of the list represents the probability that the student answer incorrectly, and the right element represents the probability that the student answered correctly. We compared the output data with the real label to compute the loss function. The loss function is defined by using binary cross entropy and the specific formula is as follows:

$$\mathcal{L} = -\sum_{t=1}^{T}[r_t' \cdot \log r_t + (1 - r_t') \cdot \log(1 - r_t)] \quad (7)$$

TABLE 2. A description of partial features shown in this paper.

ID	Feature	Description
1	user_id	The ID of the student
2	problem_id	The ID of the problem
3	skill_id	ID of the skill associated with the problem
4	ms_first_response	The time for the student's first response
...	...	...
21	opportunity	Numbers of time the student practices on this skill
22	hint_count	Numbers of student uses the hint on this problem
23	first_action	The type of first action: attempt or ask for a hint
24	attempt_count	Number of student attempts on this problem

At the t-th time, $r_t^{'}$ is the actual score and r_t is the predicted score on exercise through MFA-DKT framework. In addition, MFA-DKT framework minimizes the loss function utilize Adam optimization algorithm [25].

IV. EXPERIMENTS

First, this part introduces the experimental dataset and the parameter setup of our framework. Then, we compare our framework with several methods and verify the validity of our framework through experiments.

A. EXPERIMENTAL DATASET

ASSISTments[1] is a free learning platform used to assign math homework and classwork to students and provide feedback information to teachers. ASSISTment 2009-2010 is a dataset collected by the ASSISTments intelligent tutoring systems (ITS) [26]. The online dataset is publicly available, and has been used extensively by researchers studying knowledge tracing. In this work, we use Skill-Builder data 2009-2010[2] to conduct our experiments. We obtain 338,001 logs consisting of 4,216 students and 24,896 exercises for this dataset.

When processing the data, we remove features with a large number of null values, duplicates and texts, leaving us with 24 features. Table2 presents descriptions of the features. A more detailed description of the features is available on this website.[3]

B. EVALUATION METRIC

We evaluate MFA-DKT as the task of both classification and regression on student performance prediction. This paper selects the standard evaluation metric, including area under an ROC curve (AUC), prediction accuracy (ACC) and mean absolute error (MAE) and root mean square error (RMSE). The task of forecast performance is considerd as a classification problem, where students answer correctly as positive samples and in which negative marks are a negative sample. AUC and ACC are used as the standard of the evaluation model, and the values of AUC and ACC range from 0 to 1, with 0.5 indicating that the result of the prediction is random. A greater value indicates better performance. The prediction task also is regarded as a regression problem. MAE and RMSE are used to measure the spacing between real samples and predicted results and smaller values denote better results.

C. BASELINES

To demonstrate the effectiveness of our framework (MFA-DKT) in student performance prediction, our method compare with four methods, including Bayesian Knowledge Tracing (BKT), Deep Knowledge Tracing (DKT), Addressing Two Problems in Deep Knowledge Tracing via Prediction-Consistent Regularization (DKT+) and Dynamic Key-Value Memory Networks for Knowledge Tracing (DKVMN).

BKT : BKT is a classical method of knowledge tracing which considers students' mastery status as a binary variable. Based on the exercise sequence of corresponding concepts, BKT uses hidden Markov model to update the probability of students' mastery level.

DKT : DKT utilizes the recurrent neural network model to estimate students' performance. It takes time sequence into account to predict student performance.

DKT+ : DKT+ combines the reconstruction and waviness of regularization term with the objective function based on DKT to address the reconstruction of sequence problem and the wavy transition in prediction result which arise in DKT model.

DKVMN : Based on memory-augmented neural networks, DKVMN designs a static key matrix and a dynamic value matrix.

D. EXPERIMENTAL RESULTS

In experimental stage, we randomly select 60%,70%,80%, 90% of the sequences from dataset as training data and the remains of dataset as testing data. All experiments are repeated 10 times and selected the average experimental results as metrics. Figure 3 shows the result of our framework.

In our framework, the number of hidden units is set to 16 for LSTM network. During the model training by using Adam algorithm, the initial learning rate is set as 0.05 and learning decay rate is set as 0.0005. Moreover, the number of iterations is set as 500. Our framework use Pytorch to implement, and running environment is ubuntu server.

The experiments comparing the ACC, AUC, MAE and RMSE results of our MFA-DKT framework with four other methods: BKT, DKT, DKT+, IRF-DKT and DKVMN. Figure 3 shows the results of all of the methods. Figure 3 indicates that traditional model such as BKT is not as effective as deep learning models for student performance prediction. BKT achieved an AUC of 0.68 and an ACC of 0.58 at its best. DKT achieved an AUC of 0.85 and an ACC of 0.79,

[1] https:// www.assistments.org/

[2] https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010-data/skill-builder-data-2009-2010

[3] https://sites.google.com/site/assistmentsdata/home/assistment-2009-2010-data

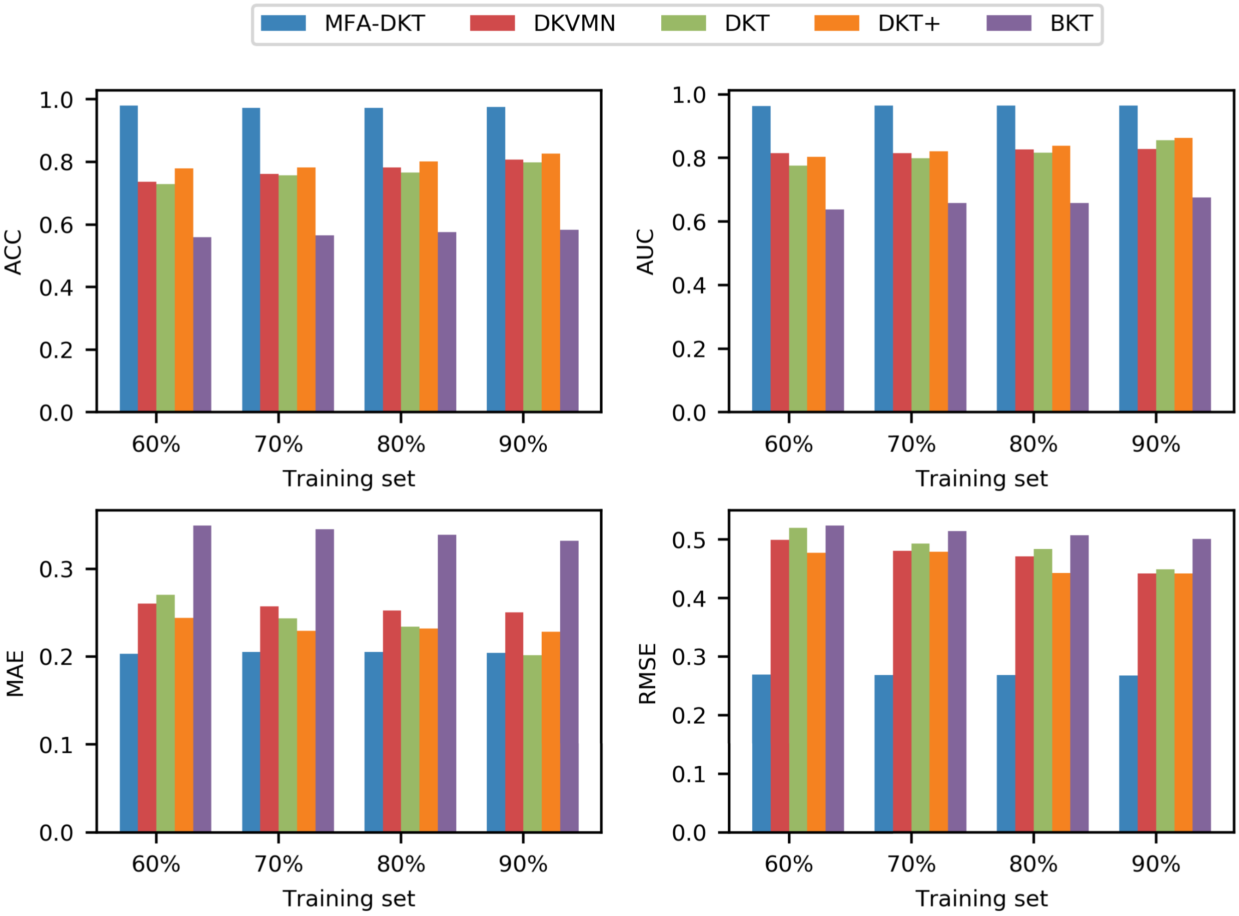


FIGURE 3. The results of student performance prediction under four metrics.

respectively. DKT+ solves the reconstruction and waviness of the regularization term with the objective function based on DKT. DKT+ produces an AUC value of 0.86 and ACC value of 0.82, which is better than that of the DKT model. The DKVMN integrates the knowledge concepts that have better performance than the DKT model. The AUC of the DKVMN is 0.83, and its ACC value is 0.81. All of the above experimental results are the best results for each model at 90% of the training set. Obviously, our framework is superior to these methods for four evaluation criteria. The experimental results of our method, in the best case, can reach values of 0.98 for AUC, 0.97 for ACC, 0.26 for RMSE and 0.20 for MAE. In summary, the experimental results demonstrate the accuracy and effectiveness of our framework in exploiting rich information of the features fusion attention mechanism.

For MFA-DKT, this paper considers the impact of two important parameters on the final performance, including the number of hidden units and iterations for the LSTM network. Figure 4a shows the performance of MFA-DKT with an increase in the number of hidden units. When the number of hidden layers is set to 16, the performance of our framework is better. Thus, we set the number of hidden units to 16. Figure 4b, as the iteration proceeds, the effect of our framework tends to increase and stabilize.

E. THE IMPORTANCE OF MULTIPLE FEATURES AND THE ATTENTION MECHANISM

This section verifies the importance of the features and attention mechanism in the model. DKT does not consider features of students; IRF-DKT considers only a few features. MF-DKT considers all features of students and MFA-DKT combines the attention mechanism based on MF-DKT. Here, our method compares DKT and IRF-DKT by selecting 70% of the data as training data, and the specific experimental results are shown in table 3.

Table 3 shows that our method with incorporated features outperform the original DKT and IRF-DKT model. In the ASSISTments 2009 dataset, the ACC value of the MF-DKT method is 0.92 and the AUC value is 0.94 after adding multiple features of students. Moreover, the fusion attention mechanism approach works better than only considering feature one. The ACC value of the MFA-DKT framework is 0.97 and the AUC value is 0.96 after adding

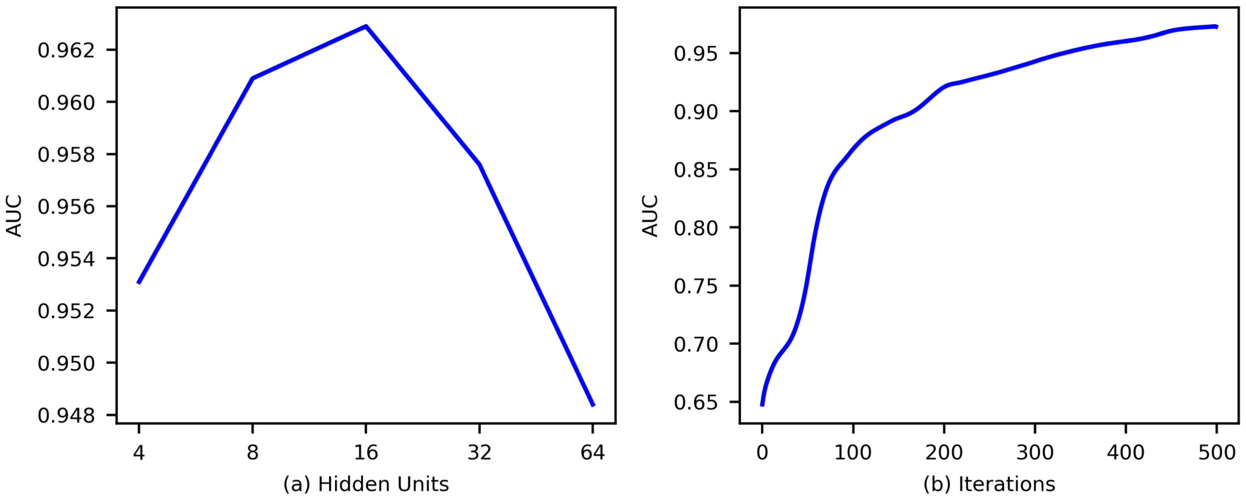


FIGURE 4. The impact of hidden units and iterations.

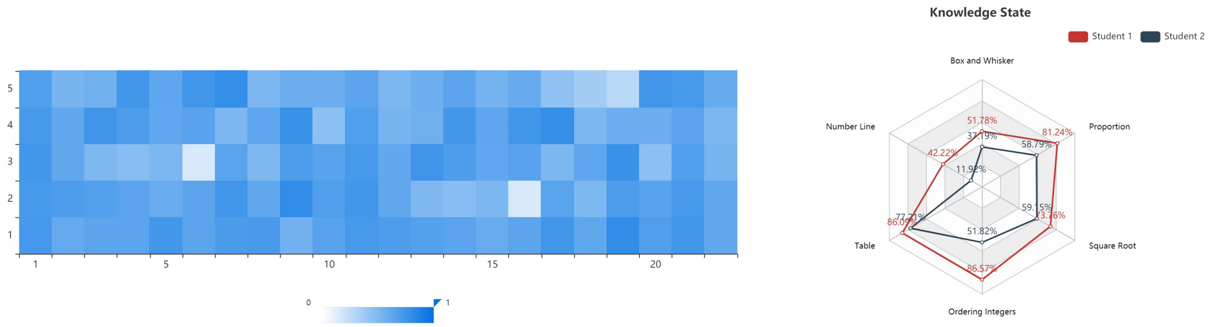


FIGURE 5. Left side shows the average mastery level of students on all concepts. Right side shows an example of the knowledge mastery level tracing of two students on 6 concepts.

TABLE 3. The experimental results compared with DKT and IRF-DKT.

Method	ACC	AUC	MAE	RMSE
DKT	0.76	0.79	0.25	0.49
IRF-DKT	0.89	0.87	0.24	0.31
MF-DKT	0.92	0.94	0.22	0.28
MFA-DKT	0.97	0.96	0.20	0.26

the multiple features fusion attention mechanism. Integrating multiple features into the knowledge tracking model can improve the accuracy of predicting student performance. Meanwhile, the attention mechanism gives the model the distinguishing capacity to improve its effectiveness. In summary, the above evidence demonstrates that our framework has a better ability to predict student performance by making full use of the multiple features fusion attention mechanism.

F. EDUCATIONAL APPLICATIONS

Our framework has several applications, such as providing feedback to teachers and creating individual learning schemes for students. For example, when students take an online course at home, teachers are not able to visualize the level of knowledge students acquire. By predicting student states to analyze students' overall knowledge weaknesses, teachers can better understand students' proficiency. As shown in Figure 5, the left side presents the overall mastery level of students on 110 concepts. The lighter the color, the worse the students' mastery level. According to the heatmap, teachers can spend more time on the knowledge points students struggle with. The right side of Figure 5 shows the knowledge mastery level of two students on 6 concepts. Students can recognize their poor knowledge points and make individual learning schemes by tracing the knowledge state.

The quality of the knowledge tracing model is measured by the accuracy of the prediction results and the availability to describe students' knowledge states [27]. Predicting student performance and describing students' knowledge states solve practical issue in the domain of education. From the above experiments, our prediction accuracy validates the effectiveness of our framework. Identifying students' knowledge states can help realize their learning situation in time. Figure 5 shows the knowledge mastery level of two students in corresponding knowledge concepts. In summary, the above evidence demonstrates that MFA-DKT has a better ability and power for student performance prediction.

V. CONCLUSION

This paper proposes a novel framework Multiple Features Fusion Attention Mechanism Enhanced Deep Knowledge Tracing (MFA-DKT), which incorporates multiple characteristics from students to predict their future performance. A method, PCA, is used to handle the students' behavior and exercise features. Then, we utilizes recurrent neural network to encode the student state and combine the attention mechanism for student performance prediction. Fully considering and analyzing the impact of student features on their knowledge state fusion attention mechanism improves the accuracy of prediction to a large extent.

First, compared with other methods, MFA-DKT greatly improves the accuracy of student performance prediction and can recommend more appropriate learning programs for students. Second, our method obtains students' learning state from their behavior features and exercise features by considering more effect of students' individuality on student performance prediction. Third, instead of selecting features by manual from subjective perceptions, this paper uses PCA to deal with the student features for comprehensively analyzing knowledge states of students. Moreover, our framework combines the attention mechanism in neural networks to consider each student feature has different importance on student performance. In conclusion, our method takes into account more factors and improves prediction accuracy compared with previous methods.

However, this work ignores the impact of the relation between knowledge concepts on student performance prediction. In the future work, we would like to consider the relation between knowledge concepts for capturing knowledge structure in prediction.

REFERENCES

[1] Y. Su, Q. Liu, Q. Liu, Z. Huang, Y. Yin, E. Chen, C. Ding, S. Wei, and G. Hu, "Exercise-enhanced sequential modeling for student performance prediction," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 2435–2443.

[2] N. Thai-Nghe, T. Horváth, and L. Schmidt-Thieme, "Factorization models for forecasting student performance," in *Proc. Educ. Data Mining*, 2010, pp. 11–20.

[3] C.-K. Yeung and D.-Y. Yeung, "Incorporating features learned by an enhanced deep knowledge tracing model for stem/non-stem job prediction," *Int. J. Artif. Intell. Educ.*, vol. 29, no. 6, pp. 1–25, 2018.

[4] L. DiBello, L. Roussos, and W. Stout, "Review of cognitively diagnostic assessment and a summary of psychometric models," in *Handbook of Statistics*, vol. 26, C. Rao and S. Sinharay, Eds. Amsterdam, The Netherlands: Elsevier, 2007, pp. 970–1030.

[5] 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, 1995.

[6] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein, "Deep knowledge tracing," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 505–513.

[7] 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.

[8] L. Zhang *et al.*, "Incorporating rich features into deep knowledge tracing?" in *Proc 4th ACM Conf. Learn.*, Cambridge, MA, USA, Apr. 2017, pp. 169–172.

[9] Q. Liu, Z. Huang, Y. Yin, E. Chen, H. Xiong, Y. Su, and G. Hu, "EKT: Exercise-aware knowledge tracing for student performance prediction," *IEEE Trans. Knowl. Data Eng.*, early access, Jun. 24, 2020, doi: 10.1109/TKDE.2019.2924374.

[10] S. E. Embretson and S. P. Reise, *Item Response Theory*. Boca Raton, FL, USA: Psychology Press, 2013.

[11] J. de la Torre, "DINA model and parameter estimation: A didactic," *J. Educ. Behav. Statist.*, vol. 34, no. 1, pp. 115–130, Mar. 2009.

[12] M. V. Yudelson, K. R. Koedinger, and G. J. Gordon, "Individualized Bayesian knowledge tracing models," in *Proc. Int. Conf. Artif. Intell. Edu.* Memphis, TN, USA: Springer, 2013, pp. 171–180.

[13] S. Cheng and Q. Liu, "Enhancing item response theory for cognitive diagnosis," 2019, *arXiv:1905.10957*. [Online]. Available: http://arxiv.org/abs/1905.10957

[14] F. Wang, Q. Liu, E. Chen, Z. Huang, Y. Chen, Y. Yin, Z. Huang, and S. Wang, "Neural cognitive diagnosis for intelligent education systems," 2019, *arXiv:1908.08733*. [Online]. Available: http://arxiv.org/abs/1908.08733

[15] J. Zhang, X. Shi, I. King, and D.-Y. Yeung, "Dynamic key-value memory networks for knowledge tracing," in *Proc. 26th Int. Conf. World Wide Web*, Apr. 2017, pp. 765–774.

[16] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, *arXiv:1412.3555*. [Online]. Available: http://arxiv.org/abs/1412.3555

[17] P. Chen, Y. Lu, V. W. Zheng, and Y. Pian, "Prerequisite-driven deep knowledge tracing," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2018, pp. 39–48.

[18] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

[19] C.-K. Yeung and D.-Y. Yeung, "Addressing two problems in deep knowledge tracing via prediction-consistent regularization," in *Proc. 5th Annu. ACM Conf. Learn. at Scale*, Jun. 2018, pp. 1–10.

[20] Y. Chen, Q. Liu, Z. Huang, L. Wu, E. Chen, R. Wu, Y. Su, and G. Hu, "Tracking knowledge proficiency of students with educational priors," in *Proc. ACM Conf. Inf. Knowl. Manage.*, Nov. 2017, pp. 989–998.

[21] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*. [Online]. Available: http://arxiv.org/abs/1409.0473

[22] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, and Y. Bengio, "Attention-based models for speech recognition," *Comput. Ence*, vol. 10, no. 4, pp. 429–439, 2015.

[23] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," *Comput. Sci.*, vol. 2015, pp. 2048–2057, Feb. 2015.

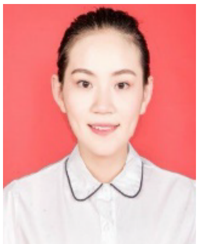
[24] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics Intell. Lab. Syst.*, vol. 2, nos. 1–3, pp. 37–52, 1987.

[25] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980

[26] M. Feng, N. Heffernan, and K. Koedinger, "Addressing the assessment challenge with an online system that tutors as it assesses," *User Model. User-Adapted Interact.*, vol. 19, no. 3, pp. 243–266, Aug. 2009.

[27] J. Lee and D.-Y. Yeung, "Knowledge query network for knowledge tracing: How knowledge interacts with skills," in *Proc. 9th Int. Conf. Learn. Analytics Knowl.*, Mar. 2019, pp. 491–500.

DONG LIU received the M.S. degree in computer science from Zhengzhou University, in 2004, and the Ph.D. degree in computer science from Tianjin University, in 2013. He is currently an Associate Professor of computer science with Henan Normal University. His research interests include educational data mining and complex network analysis.



YUNPING ZHANG received the B.S. degree in computer science and technology from Henan Normal University, Xinxiang, China, in 2018, where she is currently pursuing the M.S. degree. Her current research interest includes educational data mining.

JUN ZHANG received the Ph.D. degree from Zhengzhou University, in 2007. He is currently an Associate Professor of mechanical engineering with Zhengzhou University. His research interests include the Internet of Things technology, data mining, and complex network analysis.

QINPENG LI received the B.S. degree in computer science and technology from Henan Normal University, Xinxiang, China, in 2019, where he is currently pursuing the M.S. degree in software engineering. His current research interests include graph neural networks and educational data mining.

CONGPIN ZHANG received the B.S. and M.S. degrees in computer applications from the University of Science and Technology of China, in 2002, and the Ph.D. degree in education from Tarlac State University, in 2013. She is currently a Professor of computer science with Henan Normal University. Her research interests include language processing technology and data processing for education.

YU YIN received the B.S. degree in computer science from the University of Science and Technology of China (USTC), China, in 2017, where he is currently pursuing the M.S. degree with the School of Computer Science and Technology. His main research interests include data mining, intelligent education systems, and image recognition.