Journal of Electronics and Information Science (2024)
Clausius Scientific Press, Canada
DOI: 10.23977/jeis.2024.090102
ISSN 2371-9524 Vol. 9 Num. 1

Graph Convolutional Neural Network Knowledge Tracking Based on Response Time Feature

Juwei Dao[1,a,*], Li Hong[1,b]

[1]*College of Information, Yunnan Normal University, Kunming, China*
[a]*804079054@qq.com*, [b]*651035658@qq.com*
[*]*Corresponding author*

Keywords: Knowledge Tracking, Deep Learning, Graph Convolutional Neural Network

Abstract: The GCKT model is proposed based on the following two feature optimization ideas. Firstly, Graph Convolutional Neural Network (GCN) is applied to knowledge tracking in order to enhance local features, improve the effect of the model, and reduce the risk of overfitting. In addition, in order to solve the problem that the current GKT model only depends on the relevant content of the learner's answer and input few features, which leads to low prediction accuracy, the model in this paper uses the time features obtained by incorporating the learner's answer time of each exercise, and gives the learner each answer record as the model input.To improve the accuracy of prediction. Finally, the effectiveness and rationality of the proposed method are proved by experiments.

1. Introduction

Knowledge tracking is based on a student's time series of exercise records$[x_1, \dots, x_t]$to predict $yt \simeq xt + 1$, where$x_t = \{q_t, r_t\}$is a tuple, q_t represents an exercise that answered, and r_t represents whether the exercise answered correctly or not.[1] Deep Knowledge Tracing (DKT) [2] is the first deep learning-based method that has shown significant performance compared to traditional Bayesian knowledge tracing [3].However, DKT employs a simple Recurrent Neural Network (RNN)[4]architecture, which leads to two drawbacks: 1) It uses a hidden vector to represent the temporal knowledge state of students, which complicates modeling the knowledge state of each concept separately. 2) The simple model architecture that embeds input vectors and propagates them to recurrent layers makes it difficult to model or reflect complex relationships between concepts. Zhang et al.[5]'s Dynamic key-value Memory network (DKVMN), which utilizes two memory matrices and a simple attention mechanism to overcome these shortcomings, nevertheless, the model cannot solve complex and multiple relationships.

Inspired by Graph Neural Networks (GNN), In this paper, Nakagawa et al.[6]proposed a Graph-based Knowledge Tracing (GKT) model.They reformulated the knowledge tracing task as a classification problem of Graph time series nodes, which was the first time to be applied to the field of knowledge tracing and achieved good performance improvement. In addition, it enhances the interpretability of the knowledge tracing model, which is lacking in most other deep learning knowledge tracing models. However, GKT still does not reach an ideal level in terms of prediction accuracy.

In order to solve this problem, we propose to add a new feature of exercise answering time to the original input, so as to obtain a more accurate representation of students 'mastery level of each knowledge concept, so as to improve the accuracy of predicting students 'answer results. Accurate prediction can help students understand the true state of their learning, so as to improve the learning efficiency.

Based on the following two feature optimization ideas, this paper proposes the GCKT model. Firstly, Graph Convolutional Network (GCN)[7] is applied to knowledge tracking in order to enhance local features, improve the effect of the model, and reduce the risk of overfitting. In addition, in order to solve the problem that the current GKT model only depends on the relevant content of the learner's answer and input few features, which leads to low prediction accuracy, the model in this paper uses the time features obtained by incorporating the learner's answer time of each exercise, and gives the learner each answer record as the model input. To improve the accuracy of prediction.

2. Data Set And Knowledge Concept Analysis

In this section, the data set used by the model is first introduced, and then the information in the data set is analyzed, the problematic content in the data set is removed, and the subject features required by the model in this paper are selected. According to the previous research, it is shown that the deep knowledge tracking model can learn the substructure of knowledge concepts autonomously, and provide help for future prediction.

2.1. Dataset introduction

The model is verified on two data sets. The first data set is a public data set, "skill-builder", which is an online exercise data set of mathematics courses from 2009 to 2010 provided by ASSISTments[8], an online education platform. ASSIST2009 is one of the most commonly used datasets in the field of intelligent education, because it has the advantages of large scale, rich information, and sufficient annotation of knowledge concepts. Therefore, all intelligent education models will have relatively good performance on this data set, but it is difficult to have such a rich information scene in actual use.

The second data set is our self-made data set. The data set contains the real classroom exercise records of 414 learners in the information technology course of the eighth grade in 2023 of a junior high school in Yunnan Province, and we name the data set DH-jhs2023.

From these datasets, we select the features required by the proposed model in terms of input, including; Learner ID, knowledge concept ID, exercise question ID, correct answer or not, answer time.

2.2. Preprocess the datasets

We preprocessed the ASSIS2009 dataset as follows; (1) remove all the records without annotated knowledge point names; (2) 92 important knowledge points were selected and all records except those were removed; (3) Delete all the records under the id of learners with less than 10 records; (4) Delete all the records under the id of the learner with more than 60 records. The processed data set had 39,075 test-taking records of 1585 learners. We preprocessed the DH-jhs2023 dataset as follows; (1) manually annotate knowledge points for all test records; (2) Delete the test records with missing data (3) delete all records under the learner id name with less than 10 test records; (4) Delete all the records under the id of the learner with more than 30 records. The data set after processing had a total of 12420 test-taking records of 414 learners.

3. Model Construction

Firstly, we assume that the knowledge concept involved in the course can be defined as a graph: $G = (V, E, A)$. Where V is a node in the graph and $V = \{v_1, \ldots, v_N\}$, which represents the N knowledge concepts that the syllabus requires learners to master in the course. And E stands for edge, which is the connection between these nodes, $E \subseteq V \times V$。We use an adjacency matrix $A \in R^{N \times N}$ to define how closely each edge connects two nodes. Adding this graph structure property of knowledge as relational induction bias into the knowledge tracing model can improve the performance and interpretability of the model [9].

Secondly, we apply Graph Convolutional Neural Network, a graph-based deep learning model that generalizes the convolution operation from traditional data (images or grids) to graph data, to the knowledge tracing model. The key is to learn a function f,to generate the representation of node vi by aggregating its own features x_i and the features of its neighbors x_j, $j \in N(v_i)$, where, and complete the task of mining and predicting the hidden learning state through these representations.

Different from the traditional convolutional neural network, the convolutional layer of GCN uses the adjacency matrix and degree matrix to define the structure of the graph, so as to realize the convolution operation of the node features on the graph. In addition, GCN can train even if only few nodes have labels, which is called semi-supervised classification method, which can better adapt to the cold start problem commonly existing in the actual situation.

Next, we introduce a new feature, answer time, which will play its role in the construction of the graph. The operation is as follows. After special normalization, we map each question to the interval [0.25, 1.25] to form a weight value representing the answer time feature, which is then added to the input feature of GCN as the weight of each node vector V_i.

Finally, we assume that a learner has the knowledge state of a single knowledge concept $i \in V$ at some time t as $h^t = \{h^t_{i \in V}\}$.The update mechanism of h^t's knowledge state over time is as follows: when the learner performs an exercise of knowledge concept V_i, the learner's knowledge state in knowledge concept V_i is updated.At the same time, the knowledge state of A group of nodes N_i adjacent to V_i is also updated according to the weight in the adjacency matrix A. In general, when a student answers a concept, GCKT will first summarize the node features related to the answered concept, and then update their knowledge state. Finally, it will predict the probability of the student answering each concept correctly in the next time step.

3.1. Structure of the Model

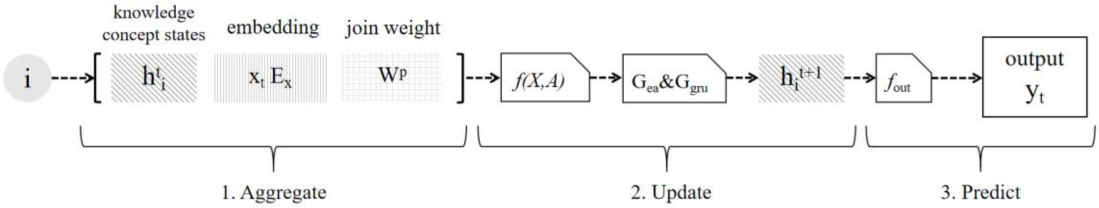

Figure 1: Graph Convolutional Knowledge Tracing (GCKT) model structure.

3.1.1. Aggregation

First, we aggregate the hidden states and embeddings of the concept i and add the answer time features to them.

$$h_i^t = x_t \, ExW^p \tag{1}$$

$$W_i^p = \sum_{i=0}^{n}[(t_i - t_{min})/(t_{max} - t_{min})] + 0.25 \tag{2}$$

W^pIs the proficiency value weight calculated by the answer time, such as the formula, $x_t \in \{0,1\}^{2N}$is an input vector, representing the correct or wrong answer of each exercise, $E_X \in R^{2N \times e}$is a matrix embedded with the knowledge concept of the question and the answer result, e is the size of the embedding.

3.1.2. Update

Next, the model will update the hidden state according to the characteristics of aggregation and the construction of knowledge graph, which is an L-layer GCN based on hierarchical propagation rules, and the propagation mode between layers is as follows.

$$h^{(l+1)} = \sigma(\widetilde{D}^{-\frac{1}{2}} \widetilde{A} \widetilde{D}^{-\frac{1}{2}} h^l W^l) \tag{3}$$

$$\widetilde{A} = A + I_N \tag{4}$$

$$\widetilde{D}_{ii} = \sum_j \widetilde{A}_{ij} \tag{5}$$

$$H^{(l)} = \begin{cases} H^{(0)} = X \\ H^{(l)} = Z \end{cases} \tag{6}$$

$$Z = f(X, A) = \sigma(\widetilde{A}XW^l) \tag{7}$$

The $\widetilde{A}$ as shown in the formula (4) is adjacency matrix A plus I_N, where I_N is an n-dimensional identity matrix. The reason we need an identity matrix is that the adjacency matrix, A has 0 on its diagonal (nodes don't have a self-loop), so when we multiply it with the feature matrix H, we ignore the features of the node itself. Adding an identity matrix I_N to A changes the diagonal entries to 1. $\widetilde{D}$ is the degree matrix for $\widetilde{A}$ and the formula is (5). $\widetilde{A}$ is an unnormalized matrix, multiplying by H will change the original distribution of the features. Therefore, it needs to be normalized so that each row of adds up to 1, become a symmetric and normalized matrix. In summary, the forward propagation formula of GCN in GCKT is as (7), W^l is the weight matrix of layer l.σ()Represents the nonlinear activation function. $H^{(l)} \in \mathbb{R}^{N*D}$ is the activation matrix of layer l. The subsequent G_{ea} is the erasure add gate used by Zhang et al[5]. $G_{gru}$is the gated recurrent unit (GRU) gate[10].

3.1.3. Predict

Finally, the model predicts the correct rate of each knowledge concept exercise in the next time the learner does the exercise, and the formula is:

$$y_k^t = f_{out}(h_k^{t+1}) = \sigma(W_{out} h_k^{t+1} + b_k) \tag{8}$$

W_{out} is a weight matrix shared by all nodes, b_k is an offset for node k, and σ is a sigmoid activation function. The proposed model uses the negative log-likelihood function (NLL) as the loss function as shown in Equation (9), which represents the difference between the true value and the predicted value, and updates the model based on it.

$$L(\theta) = -\sum_{i=1}^{m} \log(P(x^i; \theta)) \tag{9}$$

4. Performance Evaluation

4.1. Experiment details

During training, we set some important hyperparameters of the model as shown in Table 1.

Table 1: Experimental hyperparameter Settings.

Parameters	Settings	Description
hid-dim	32	Hide the dimension of knowledge states
emb-dim	32	Dimensions of knowledge concept embedding
epochs	150	Training iteration limit
batch-size	128	Number of samples per batch
optimizer	Adam	Optimizer selection

4.2. Experimental Results

Generally, the main measures of model performance in the same type of task are Accuracy (ACC) and Area under Curve (AUC), where, the Curve in question is the Receiver Operating Characteristic Curve (ROC). This study uses the above two methods to evaluate the performance of our machine learning algorithm.

We evaluate the predictive performance of GCKT,and compare with the Commonly used models —— DKT and GKT. The results are shown in the Table 2:

Table 2: Experimental comparison results.

Method	AUC		ACC	
	ASSIS2009	DH-jhs2023	ASSIS2009	DH-jhs2023
DKT	0.757	0.752	0.707	0.748
GKT	0.768	0.759	0.721	0.746
GCKT	0.790	0.773	0.732	0.767

Experiments show that the AUC of the GCKT model proposed in this paper is improved in the public data set ASSISTMents2009 and the self-made data set DH-jhs2023. Due to the improvement of the input characteristics, the GCKT model proposed in this study demonstrates excellent knowledge tracking ability.

Deep learning techniques such as LSTM and DKT have been widely used in natural language processing tasks. However, this paper proposes a novel graph neural network-based method called GCKT, which has more advantages over the traditional DKT. Through the comparative study of different algorithms, it is found that GCKT can significantly improve the effect of knowledge tracking and enhance its interpretability. In addition, we also note that compared with the traditional long Short-Term memory network (LSTM) and GKT methods, GCKT achieves better performance under the novel information acquisition method. This is mainly because GCKT uses the newly introduced temporal features to better simulate the actual scene, thus achieving better knowledge tracking ability.

5. Conclusion

In this paper, the GCKT model is proposed. By analyzing the log of answer data, considering the realistic situation of learners, the time information of learners' historical answer questions is integrated into the knowledge tracking model, and then Graph Convolutional Neural Network (GCN) is used to enhance the features to obtain better knowledge tracking effect. In this paper, the design of the GCKT model, the Settings of various parameters and the corresponding applications, the data input methods, the improvement of the model algorithm, and the output of the model are introduced in detail. Through the improved model, the performance of the knowledge tracking model is further improved, and a learner model with good performance is provided for the following exercise recommendation system. The main work of the model is as follows:

(1) Based on the analysis of GKT and the data observation of learners' answer records in the dataset, it is found that the accuracy of the original GKT model is not ideal due to the lack of features. Therefore, GCKT is proposed to optimize the features of GKT.

(2) Compared with the standard GKT model, GCKT first applies GCN to knowledge tracking to enhance local features, improve the effect of the model, and reduce the risk of overfitting. In addition, the learner's answer time information is added, and the weight increment calculated by the answer time is added to the graph node construction, so as to optimize the input features to improve the accuracy of the improved model prediction.

(3) By conducting multiple rounds of experiments, we analyze the AUC and ACC values of various models on the test set in detail, which proves that our GCKT model with feature optimization can effectively use many features, thereby improving the performance of the model.

References

[1] Khajah, M., Lindsey, R. V., & Mozer, M. C. (2016). How deep is knowledge tracing?. arXiv preprint. DOI:10.48550/arXiv.1604.02416.

[2] Piech, C., Spencer, J., Huang, J., Ganguli, S., Sahami, M., & Guibas, L., et al. (2015). Deep knowledge tracing. Computer Science, 3(3), págs. 19-23.DOI:10.1109/CVIDLICCEA56201.2022.9823984.

[3] Albert, T., CorbettJohn, R., & Anderson. (1994). Knowledge tracing: modeling the acquisition of procedural knowledge. User Modeling & User Adapted Interaction.DOI:10.1007/BF01099821.

[4] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.DOI:10.1162/neco.1997.9.8.1735.

[5] Zhang, J., Shi, X., King, I., & Yeung, D. Y. (2017). Dynamic key-value memory networks for knowledge tracing. International World Wide Web Conferences Steering Committee, 765-774. DOI:10.1145/3038912.3052580.

[6] Nakagawa, H., Iwasawa, Y., & Matsuo, Y. (2019). Graph-based Knowledge Tracing: Modeling Student Proficiency Using Graph Neural Network. IEEE/WIC/ACM International Conference on Web Intelligence. ACM. DOI: 10.1145/3350546.3352513.

[7] Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint. DOI: 10.48550/arXiv.1609.02907.

[8] Feng, M., Heffernan, N., & Koedinger, K. (2009). Addressing the assessment challenge with an online system that tutors as it assesses. User Modeling and User-Adapted Interaction, 19(3), 243-266. DOI:10.1007/s11257-009-9063-7.

[9] Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., & Malinowski, M., et al. (2018). arXiv preprint. Relational inductive biases, deep learning, and graph networks. DOI:10.48550/arXiv.1806.01261.

[10] Cho, K., Van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., & Schwenk, H., et al. (2014). Learning phrase representations using rnn encoder-decoder for statistical machine translation. Computer Science. DOI: 10.3115/v1/D14-1179.