International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 776-788

https://doi.org/10.70517/ijhsa46471

Research on Strategies for Improving Students' Independent Learning Ability in English Teacher Training Empowered by Artificial Intelligence Algorithms in the New Era

Qiaoli He[1] and Yadan Deng[2,*]
[1] Xiangnan University, Chenzhou, Hunan, 423000, China
Corresponding authors: (e-mail: 13807350979@163.com).

Abstract As the backbone of future basic education, the independent learning ability of English teacher trainees directly affects the quality of teaching and student development. However, the existing cultivation mode lacks personalized guidance, and it is difficult to accurately identify the learning state, resulting in poor cultivation effect. Focusing on the cultivation needs of independent learning ability of English teacher training students, this study constructs a deep knowledge tracking algorithm based on students' state (DKT-ST) and an open learner model (ALS-OLM) for independent learning services. The model adopts a two-layer memory network structure, introduces the forgetting factor and attention mechanism, stores knowledge concept information through a static matrix, and updates students' knowledge mastery through a dynamic matrix to realize accurate tracking of learning status. Experiments on three public datasets show that the AUC value of the DKT-ST model on the ASSISTments2009 dataset reaches 0.817, which is a 5.42% improvement compared with the DKT-DSC model, and on the KDDCup2010 dataset is a 2.21% improvement. The results of the teaching experiment show that the average improvement of students' independent learning ability in the experimental class is 25.61 points, with an increase of 37.20%, which is significantly better than that of the control class, which is 14.89%. The analysis of students' ability attributes shows that the key ability dimension is improved by 0.46. The study proves that the personalized learning model based on artificial intelligence algorithms can effectively improve the independent learning ability of English teacher training students, and provides technical paths and practical references for the reform of teacher education.

Index Terms artificial intelligence algorithm, deep knowledge tracking, independent learning ability, English teacher training students, personalized learning, open learner model

I. Introduction

As early as the 1960s, Western educators began to advocate independent learning and advocate it as one of the main goals of educational reform [1]. In the 1970s, independent learning entered the field of language learning [2]. In 1982, Henri Holect published the monograph Autonomy and Foreign Language Learning which became the launching point of the research in this field. Since then, the research on autonomous learning has flourished, and with the continuous deepening and development of foreign language teaching research, how to apply the theory of autonomous learning to English learning and improve students' ability to learn English autonomously has become an important part of the current research on English learning [3]. Different schools of thought have different insights into independent learning, and some theoretical results have been achieved, but there are not many studies on the outstanding effectiveness of teacher training in English learning.

Since the 1980s, some specific teaching models have emerged internationally that are more characteristic of independent learning and more in line with the curriculum standards, mainly interactive, cooperative learning, inquiry-based learning, problem-based learning model, and strategic content-guided model [4]-[7]. Distinguished from the traditional learning mode, in independent learning learners participate in managing their whole learning process, through the planning, monitoring, evaluation and other activities of their own learning, learners hold more dominance in making decisions and taking actions, as well as taking more responsibility in the learning process, promoting independent thinking ability and improving subjective initiative [8]-[11]. However, the current English teaching is still dominated by classroom teaching, and teachers usually focus only on the explanation of teaching materials and the teaching of language knowledge, but seldom pay attention to students' learning methods and teach practical learning strategies [12], [13]. As a result, students are unable to develop good learning habits and self-learning ability. In the new era, artificial intelligence technology is widely used in the field of education, which has injected a new impetus for the improvement of students' independent learning ability [14].

In this study, we constructed an open learner model for autonomous learning services and designed a deep knowledge tracking algorithm based on student states. Firstly, the design principle and structural framework of ALS-OLM model are established, and a hybrid architecture combining independent and embedded modules is adopted to realize cross-system data sharing and privacy protection. Then the traditional deep knowledge tracking algorithm is improved, forgetting factor and attention mechanism are introduced, and the DKT-ST model is constructed to realize accurate tracking and prediction of learning state. The performance of the algorithm is verified through comparative experiments on multiple public datasets, and teaching experiments are designed to analyze the effect of the model on the enhancement of English teacher trainees' independent learning ability. The study adopts a combination of quantitative and qualitative methods to verify the effectiveness of the technical solution through pre and post-test comparative analysis.

II. Open learner model for self-directed learning services

II. A. ALS-OLM design principles

Based on the characteristics and requirements of the open learner model in terms of open content, open form and open privileges, and combined with the practical application of adaptive learning service oriented, the study follows the following design principles in the process of constructing the ALS-OLM model.

ALS-OLM should be open and shared. The model should be able to seamlessly access the cloud storage and processing system, so that users can access their data and models anytime and anywhere, enhancing the accessibility and convenience of data. The model should adopt an open and standardized data format that supports cross-system data exchange and integration, facilitates cross-platform sharing of data, and promotes operation and sharing between different systems and applications.

ALS-OLM should be flexible and adaptable. The model should adopt a hierarchical design, with each level focusing on specific functions, which can be flexibly combined according to the specific needs of users, while maintaining data circulation between levels, realizing the design principle of high cohesion and low coupling, and improving the maintainability and extensibility of the model.

ALS-OLM should be transparent and interactive. The model should provide clear data display and detailed learning records, so that learners can easily track their learning sessions and effectiveness, and can display different contents according to the authority to protect learners' privacy while meeting the needs of different learners. The model should provide real-time feedback information, support learners to interact with the model, provide personalized guidance, enhance learners' motivation for independent learning, and improve learning efficiency and satisfaction.

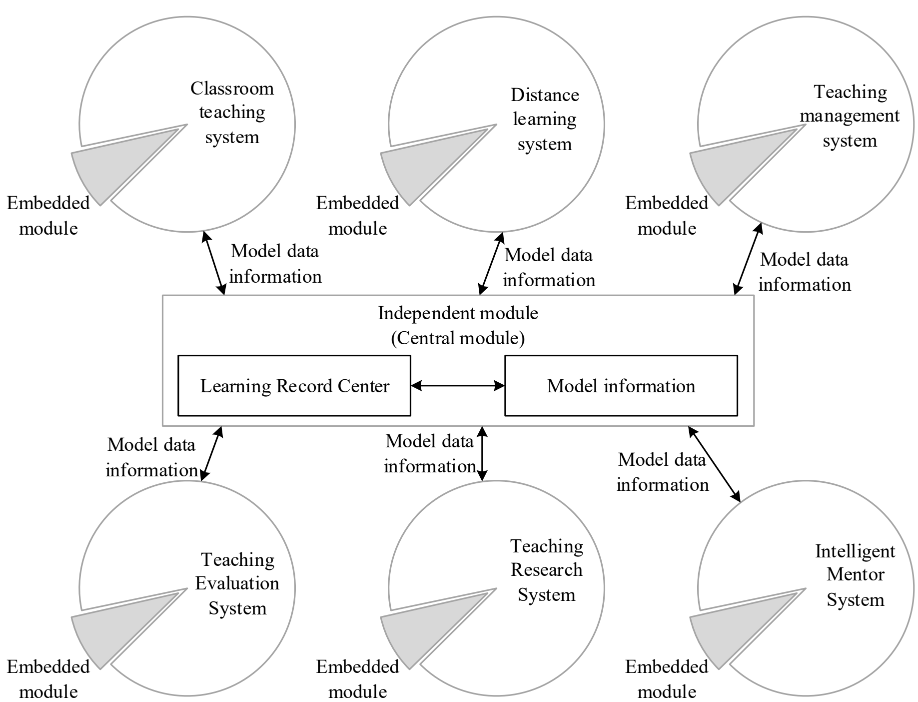

Figure 1: The relationship between the independent module and the embedded module

II. B. ALS-OLM structural design

The ALS-OLM constructed by the study combines the features of both independent and embedded models to form a hybrid model, which is divided into independent and embedded modules. The independent module can exist independently of the learning system, while the embedded module is embedded in various types of systems, such as the classroom teaching system, the distance learning system, the teaching management system, the teaching evaluation system, the teaching research system, and the intelligent tutor system, etc., which forms a star-shaped structure shown in Figure 1.

In ALS-OLM, the independent module (center module) stores data such as learners' personal basic information, academic background information and learning style information, and the learning record center can trace all learning records. When the classroom teaching system, distance learning system and other systems need to access the ALS-OLM information, they need to obtain the authorization of the learners to safeguard the privacy and security of the learners. When these systems collect a large amount of learning data that needs to be shared, the center module will first record these learning data, then judge and filter them, and update the data in the center module through algorithms.

ALS-OLM's embedded modules can be integrated into classroom teaching systems, distance learning systems, faculty management systems, teaching evaluation systems, teaching research systems, and intelligent tutor systems, provided that these systems follow the specifications of the standalone modules for data sharing. When the embedded module is launched for the first time, it will obtain the model data of the learner through the open API according to his/her authorization, and this method effectively solves the cold-start problem so as to provide more accurate services for the learning system. The embedding module is able to store the acquired model data in its own

In ALS-OLM, the model structure is divided into three layers, which are the basic data layer, the learning state data layer, and the emotion and feedback data layer. The basic data layer consists of personal basic information, which provides basic data support for the other layers. The learning state data layer consists of knowledge mastery information, learning behavior information and metacognitive ability information, providing state information for the calculation and portrayal of the open learner model. The emotion and feedback data layer brings together emotional interest social information and learning feedback information to further enhance the optimization ability of the model.

ALS-OLM is divided into static model layer and dynamic model layer. The static model layer stores learners' long-term unchanged information, such as name, gender and other basic identity information, as well as learners' long-term accumulated basic knowledge and learning styles. The static model layer provides a framework for presenting learners' basic characteristics and attributes. The dynamic model layer stores real-time information such as the learner's current learning status, learning records and test scores on the basis of the static model layer. The dynamic model layer feeds back more complex student data to the learner model by collecting and organizing the data. Mutual complementation and feedback between the static model layer and the dynamic model layer ensures that the model is continuously optimized, and the static information is fed back and corrected by the dynamic data, which enables real-time docking of the learners' long-term characteristics and short-term status, and improves the accuracy and adaptability of the model. The dynamic model layer can store specific learning details according to the needs of different disciplines or learning systems, adapting to diversified scenarios and improving the scalability of the model. The dynamic model layer continuously feeds back to the static model layer by collecting real-time learning data. A data-driven closed loop is formed, realizing self-adjustment and optimization of the model.

III. Deep Knowledge Tracking Algorithm Based on the State of English Teacher Training Students

III. A. Common algorithms for knowledge tracking

III. A. 1) Bayesian knowledge tracing

Bayesian Knowledge Tracking [15] is arguably the first model to relax the assumption of static knowledge states. Earlier approaches to knowledge tracing, such as IRT, would assume that students would not learn between answers, which is a reasonable assumption to test, but not suitable for application in a learning environment.The BKT model was introduced for knowledge tracing in a learning environment.

Specifically, the BKT model assumes that knowledge points are independent of each other, models each knowledge point individually, and uses binary variables to model the student's mastery state of the knowledge point, and updates the corresponding knowledge mastery state based on whether the student answers the question correctly or not, thus enabling prediction of the probability of the student answering the next question correctly.

The standard BKT model is composed of four parameters, which are as follows:

$P(L_0)$: denotes the initial probability that the student masters the skill L_0 ;

$P(T)$: denotes the probability that the student transitions from an un-mastered state to a mastered state;

$P(S)$: the probability of failure, denoting the probability that the student will answer incorrectly even if he or she has mastered the skill;

$P(G)$: probability of guessing, indicating the probability that the student would have answered correctly even if the skill had not been mastered.

These 4 parameters are typically learned from the student's answer record when modeling each skill.The inferred probability of the BKT model depends largely on these 4 parameters because the student's incorrect and correct attempts at the skill question to date are listed chronologically, so these parameters can be used to predict how the student has mastered a particular skill.The specific formulas for the BKT model are as follows:

$$P(L_n \mid Correct) = \frac{P(L_{n-1})(1-P(S))}{P(L_{n-1})(1-P(S))+(1-P(L_{n-1})P(G))} \tag{1}$$

$$P(L_n \mid Incorrect) = \frac{P(L_{n-1})P(S)}{P(L_{n-1})P(S)+(1-P(L_{n-1}))(1-P(G))} \tag{2}$$

$$P(L_n) = P(L_{n-1} \mid Outcome)+(1-P(L_{n-1} \mid Outcome))P(T) \tag{3}$$

The BKT model realizes dynamic tracking of students' knowledge acquisition, however, the BKT model assumes that as long as a student has mastered a certain knowledge point, the mastery level will not change at a later stage due to forgetfulness factor or any other reasons, but will keep mastering that knowledge point. In addition, the BKT model assumes that the difficulty of the exercises is the same if they contain the same knowledge points, and that the student's performance is not affected by any other factors, which limits the effectiveness of the BKT model in knowledge tracking problems.

III. A. 2) Deep knowledge tracking

Deep Knowledge Tracking is the first knowledge tracking model based on deep neural networks. Similar to the BKT model, Deep Knowledge Tracking examines students' answer sequences, but the authors take advantage of neural networks to break the limitations of skill separation and binary state assumptions, and also pioneer the use of deep learning in knowledge tracking tasks.The DKT model utilizes a student's previous answer sequences and converts each answer into a feature vector represented by a unique hot code. These features are then fed into a neural network and the information is passed through the hidden layer of the network to the output layer, which provides the predicted probability that the student will answer a particular question correctly in the system.

Specifically, the DKT model can be divided into two processes: modeling the student's knowledge state and predicting the student's performance. In the process of modeling a student's knowledge state, for a given input $x_i = (q_i, a_i)$, the model updates the student's knowledge state h_i at each time step, and the input x_i is a one-hot vector, which is later transformed into a low-dimensional dense vector v_t , using the embedding vector v_t and the previous student knowledge state vector h_{i-1} , the student's knowledge mastery state vector at a specific point in time is updated by Eq:

$$h_t = \phi(v_t, h_{t-1}) \tag{4}$$

The process of predicting student performance calculates the probability of a correct response y_i based on the updated student knowledge state h_i , which is given by the following formula:

$$y_t = \sigma(b^{out} + W^{out} h_t) \tag{5}$$

where $\sigma(\cdot)$ is the sigmod function, W^{out} is the weight matrix, and b^{out} is the output bias vector.

III. A. 3) Dynamic key-value memory networks

Inspired by memory-enhanced neural networks, Dynamic Key-Value Memory Network [16] is proposed to improve the structure of the DKT model. Due to the limited ability of hidden states in LSTM networks to represent knowledge states, memory matrices need to be introduced to store richer hidden information. The DKVMN model [17] uses two memory matrices, one of which is static for storing potential knowledge concepts of a problem, and the other is dynamic for representing the student's knowledge states, with each matrix slot storing the state of a concept. Learning or forgetting a particular skill is stored in these two parts and is controlled by additional attentional mechanisms for read and write operations. Unlike the DKT model, the DKVMN model performs local state transitions through read and write operations, avoiding global and unstructured state-to-state transitions in the hidden layer.

The DKVMN model reads and updates the dynamic matrix by calculating the relevant weights from the input questions and static matrix, which in turn updates the students' knowledge state. It consists of three main steps:

(1) Correlation weight calculation:

The correlation weights between the input questions and the knowledge points are computed using softmax activation of the inner product of k_i and $M^k(i)$ as follows:

$$w_i(i) = \text{Softmax}(k_i^T M^k(i)) \tag{6}$$

where k_i is the continuous embedding vector of the problem q_i, $\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}}$ is differentiable, and w_i represents the correlation weights between the exercises and each knowledge point, and this weight vector will be used by both the subsequent reading process and the writing process.

(2) Reading process:

The read process mainly predicts the probability of answering a newly arrived exercise correctly. When an exercise is reached, the DKVMN model calculates the student's mastery level of the exercise based on the value matrix M_i^v and the weights associated with the knowledge points of the topic w_i, which is calculated as follows:

$$r_i = \sum_i w_i(i) M_i^v(i) \tag{7}$$

The computed r_t is considered as a summary of the student's level of mastery of the exercise. Given that each exercise has its own difficulty, it is necessary to connect r_t with the vector k_t and then pass it through a Tanh-activated fully-connected layer to obtain a summarized vector f_t, with the student's mastery level and the difficulty of the exercise contained in f_t.

$$f_t = \text{Tanh}(W_1^T[r_t, k_t] + b_1) \tag{8}$$

where $\text{Tanh}(z_i) = (e^{z_i} - e^{-z_i}) / (e^{z_i} + e^{-z_i})$. Finally, f_i predicts student performance through another Sigmoid-activated fully connected layer:

$$p_t = Sigmoid(W_2^T f_t + b_2) \tag{9}$$

This gives the probability p_i that the student answered the exercise correctly, where $Sigmoid(z_i) = 1/(1 + e^{-z_i})$.

(3) Writing process:

After the student answers the question, the DKVMN model will update the value memory matrix M_t^v based on the student's answer. The input $x_t = (q_t, a_t)$ is converted to an embedding vector v_t and written to the value memory using the same correlation weights w_t as in the reading process. Before adding new information, perform an erasure of the memory using the following method:

$$e_t = Sigmoid(E^T v_t + b_e) \tag{10}$$

where the dimension of the transformation matrix E is $d_v \times d_v$, and e_t is a column vector with d_v elements, with each value taking a value in the range (0, 1). Erasure is done on the value matrix M_i^v as follows:

$$M_t^v(i) = M_{t-1}^v(i)[I - w_t(i) e_t] \tag{11}$$

where I is the row vector where all values are 1. Thus, the elements of this matrix position are reset to 0 only if the weights and erasure elements of this matrix position are both 1. If the weights or erasure signals are zero, the vector remains unchanged. After erasure, an increasing vector a_t of length d_v is used to update the value matrix M_i^v:

$$a_t = \text{Tanh}(D^T v_t + b_a)^T \tag{12}$$

where the dimension of the transformation matrix D is $d_v \times d_v$, a_t is the row vector, and the value matrix M_t^v is updated at each moment:

$$M_t^v(i) = M_{t-1}^v(i) + w_t(i) a_t \tag{13}$$

This "erase-and-add" mechanism allows students to forget and reinforce the state of knowledge points during the learning process, which is not possible with other RNN-based models.

III. B. Memory-enhanced neural networks

To address the limitations of traditional RNNs, the idea of MANN is applied to design external memory for storing long-term memory information. MANN has made progress in various fields such as question and answer systems, natural language conversion, speech recognition and video analysis.

Memory augmentation networks for knowledge tracking consider the external memory matrix of MANN as the student's knowledge state. The memory is notated as M_t and is an $M \times N$ matrix. Where M is the number of memory slots and N is the size of each memory slot. At each time t, the input to the network is the joint embedding vector v_t of the sequence of student quizzes, each time the input is one of a set of Q different question labels, and r_t is a binary value representing how the question was answered. The embedding vector v_t is utilized to compute the read weights w_t^r and write weights w_t^v. In the read mechanism, r_i is the weighted sum of the read weights w_i^r for all memory slots:

$$r_t = \sum_{i=1}^{N} w_t^r(i) M_t(i) \tag{14}$$

Then, further computation is performed to obtain the predicted value p_t, which represents the student's performance in the next training. In the write operation mechanism, the irrelevant data in the memory slots are first removed using the deletion vector e_t and the write weights w_t^w, and then the knowledge increment v_t is performed by the addition vector α_t.

III. C. Deep Knowledge Tracking Algorithm Design

The DKT-ST model is designed to model historical student-system training data and forgetting factors to track student knowledge acquisition, and its network structure is detailed in Figure 2.

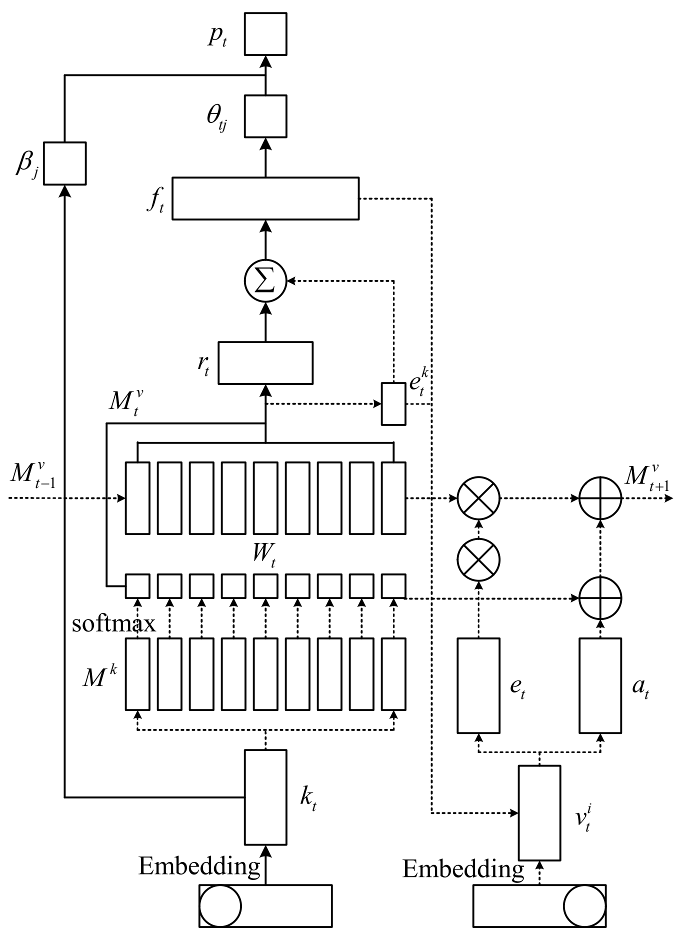

Figure 2: Deep Knowledge Tracking Algorithm Based on Student Status

The static matrix M^k stores the information about the KCs involved and the dynamic matrix M^v is used to store as well as update the students' knowledge acquisition. The algorithm is divided into three main modules.

III. C. 1) Attention mechanisms

Students at time t are used to get the title tag of q_t, will study it with network trained embedded matrix A multiply by input vector k_t, by taking k_t and each key slot $M^k(i)$ inner product between Softmax activation, Further calculation attention vector w_i, said title q_i and weight proportion between each involved KC. Both the reading mechanism and the update mechanism require the use of this vector w_t.

$$w_t(i) = Soft\max(k_t^T M^k(i)) \tag{15}$$

where $Soft\max(z_i) = e^{z_j} / \sum_j e^{z_j}$ and microscopic, $M^k(i)$ is the i th row of the static matrix M^k, $w_t(i)$ is the weight value accounted for by the i th knowledge point count, i.e., the proportion of that knowledge point in the topic q_t.

III. C. 2) Reading mechanisms

The reading mechanism is a predictive process of knowledge tracking.

This algorithm introduces the forgetting factor in the training process of the students and in this step, the forgetting vector is designed according to the knowledge mastery of the students.

$$e_t^k = Sigmoid(E_e^k M_t^v + b_e^k) \tag{16}$$

Then, drawing on the forgetting gate principle of LSTM, the dynamic matrix $M_t^{v'}$ of students' current knowledge mastery is extracted based on the input vector k_t, the weight vector w_t and the forgetting vector e_t^k.

$$M_t^{v'} = M_t^v(i)[1 - w_t(i)e_t^k] \tag{17}$$

Thus, r_i is computed as follows.

$$r_t = \sum_i w_t(i) M_t^{v'} \tag{18}$$

The knowledge state vector r_t and the input vector k_t are processed by a multilayer perceptron to obtain the summary vector f_t. This vector reflects the state of knowledge of the student and the characteristics of the question itself, such as the difficulty of the question, and shows the state of the student's aggregate knowledge of a specific question.

$$f_t = \mathrm{Tanh}(W_f^T[r_t, k_t] + b_f) \tag{19}$$

Finally, f_t utilizes the fully connected layer of the network to output the correctness of the student's next training. At this point, the reading process of the dynamic key-value memory network knowledge tracking method based on students' knowledge states has been completed.

$$p_t = Sigmoid(W_p^T f_t + b_p) \tag{20}$$

where $Sigmoid(z_i) = 1/(1+e^{-z_i})$, and p_t is a scalar representing the probability that a student correctly answers the question q_t.

III. C. 3) Updating mechanisms

The update-write mechanism of DKT-ST is used to update the dynamic matrix M^v that stores the students' knowledge mastery.

The question-answer tuple $x_t = (q_t, a_t)$ is multiplied by another embedding matrix B to obtain the knowledge growth vector v_t acquired by the student after training. To address the second problem of DKVMN, this algorithm here introduces the forgetting factor e_i^k of the student during the training process. This in turn enriches the factors involved in the student's knowledge increment.

$$v_t' = [v_t, f_t] \tag{21}$$

The process of updating the student's knowledge is similar to the forgetting gate of LSTM, in that the useless information in the dynamic matrix memory has to be deleted before the student's current knowledge growth vector can be written into the memory. In DKVMN, this process is called “erasure”.

$$e_t = sigmoid(E_e v_t^{'} + b_e) \tag{22}$$

However, from the formula, we can conclude that if it is oriented to the same student, no matter when, as long as the growth of their knowledge is the same will have the same "erase" data, which is obviously inconsistent with the actual situation.

According to the human cognitive process and forgetting theory related research, students in the long-term learning and training process, there is bound to be a certain amount of knowledge forgotten, which leads to the student's knowledge growth and the student's current degree of knowledge mastery there is a certain link. Therefore, this paper introduces the forgetting vector in the deletion vector, and adopts the linear combination of e_t^k and e_t to compute it.

$$e_t^{'} = \lambda_1 e_t + \lambda_2 e_t^k \tag{23}$$

After using the deletion vectors to remove some of the memory in the dynamic matrix M^v, the students' knowledge growth vectors are used to compute the addition vectors α_t, which is computed similarly to the LSTM.

$$\alpha_t = \tanh(D^T v_t^{'} + b_\alpha) \tag{24}$$

After obtaining the deletion vector $e_t^{'}$ and the addition vector α_t, it is possible to perform an update operation on the memory of the dynamic matrix M^v storing the students' knowledge acquisition.

$$M_t^v = M_{t-1}^v[1 - w_t(i)e_t^{'}] + w_t(i)\alpha_t \tag{25}$$

That is, the value of the dynamic matrix is transformed from M_{i-1}^v to M_i^v after the student's response behavior at the t moment. The loss function of the network uses minimizing the difference between the predicted value of the network and the true value of the response given by the student as an optimization objective, i.e., minimizing the cross-entropy of p_t and a_t. And the training process of the network is implemented by stochastic gradient descent method.

$$L = -\sum(a_t \log p_t + (1-a_t)\log(1-p_t)) \tag{26}$$

III. C. 4) Student ability and difficulty network design

Because the overall architecture of the network is relatively simple, it can be easily enhanced and extended to provide additional meaningful information during model impact. Therefore, this paper uses the knowledge status of each potential KC to calculate the student's ability. Specifically, when the network receives the KCs of the input question, it will form the eigenvector f_t during the influence period. Since f_t is a concatenation of the read vector r_t and the KCs embedding vector k_t, it contains both the information of the student's knowledge state involved in q_t and the embedding information of q_t, so the neural network can further process f_t to infer the student's ability to q_t. Similarly, the difficulty level of q_t can be obtained by passing the KCs embedding vector k_t to the neural network.

Depending on the purpose of the neural network, the two networks will be referred to as the student ability network and the difficulty network, respectively, and the two networks will be represented as such using a single fully connected layer:

$$\begin{cases} \theta_{tj} = \mathrm{Tanh}(W_\theta f_t + b_\theta) \\ \beta_j = \mathrm{Tanh}(W_\theta q_t + b_\beta) \end{cases} \tag{27}$$

The curvilinear tangent is used as the activation function for both networks in order to scale both outputs to the range (-1, 1). Then

where θ_{ij} and β_j are the student's learning ability on the topic q_t and the level of topic difficulty at time t, respectively, and the two values are passed to the item response function to calculate the probability of the student answering correctly to the question q_t using the double post.

$$p_t = sigmiod(3 - 0 * \theta_{ij} - \beta_j) \tag{28}$$

IV. Experimental analysis of knowledge tracking for English teacher training students

IV. A. Experimental data set

In this chapter, the experiments use three public online education datasets, ASSISTments2009, ASSISTments2017 and KDDCup2010.

KDDCup2010: this dataset was developed as a dataset for the 2010 KDD Cup Educational Data Mining Challenge, which was designed to predict students' performance in math problems.

ASSISTment2009: This dataset is the student learning data collected by the ASSISTments online learning platform in 2009, which includes high school math learning problems, which is a classic dataset in the field of knowledge tracking.

ASSISTment2017: this dataset is the latest dataset of ASSISTmentsData.

IV. B. Evaluation indicators

With reference to other current work in the field of knowledge tracking, the comparative experiments in this paper use the area under the curve (AUC) of the receiver operating characteristic (ROC) curve to measure the model performance, which has better performance metric properties. The output range of the AUC indicator is within the (0,1) interval, and the closer the AUC value is to 1.0, it means that the model has better performance and its prediction is better. To minimize the effect of experimental errors, for each model, this paper conducts 20 tests and calculates the average AUC value. The model is implemented using the Pytorch framework with 5-fold cross-validation to randomly select and divide the dataset.In each experiment, 60% of the students are the training set data, 20% are the test set data, and the remaining 20% are the validation set data.Each epoch is evaluated in the cross-validation set after the training is completed. The length of the learner interaction sequence in this experiment is set to 200, and the batch size is 24. Since too small a K-means clustering number will affect the experimental effect, and too large a number will increase the computational complexity, the value of 10 is determined based on the results of comparative tests under different values of K. The K-means clustering number is set at 10, which is the same as the K-means clustering number.

IV. C. Analysis of results of student performance prediction

In order to evaluate the performance of DKT-ST model, five typical benchmark models are selected in this paper, which are Bayesian Knowledge Tracking BKT, Deep Knowledge Tracking DKT two classical models as well as AKT, SAKT and DKT-DSC model based on DKT. Among them, BKT is one of the classical knowledge tracking models, which has been described in detail in Chapter 2, and is not introduced here; DKT is a classical deep learning-based model, which uses RNN modeling to learn the interaction sequences, and is a pioneering work in the field of deep knowledge tracking; in addition to the typical models, many improved models have been proposed in the field of deep knowledge tracking in recent years, and in this section, we select three models which are similar to the proposed model DKT-ST. The proposed model DKT-ST is a typical model that has similarities with the proposed model, AKT model is a knowledge tracking model based on the attention mechanism, which introduces contextual information while combining with IRT, and uses the attention mechanism to enhance the interpretability of the model; SAKT is a knowledge tracking model based on the Transformer model, which can be regarded as a kind of special case of AKT; DKT-DSC model learns the interaction sequences by the exercise Correctness indicates students' ability, which is applied to the classical DKT model. The experimental results are shown in Table 1.

Compared with the DKT-DSC model, DKT-ST, which considers students' response time and answer results for each knowledge point and combines the attention mechanism, is more effective, and the AUC index is improved by 5.42% and 2.21% on ASSISTments2009 and KDDCup2010, respectively. According to the experimental results, the deep learning-based methods are generally better than BKT, in which the DKT model only uses recurrent neural networks to construct the model, represents the learner's knowledge level as a hidden state, and predicts it in combination with the input interaction records at each moment, and does not consider the learner differences, so the prediction performance of DKT is lower than that of AKT and DKT-DSC, and the DKT-ST is more effective than AKT and DKT-DSC in terms of three datasets with the Average AUC values.

Table 1: Experimental results comparing model performance (AUC)

Dataset	BKT	DKT	SAKT	AKT	DKT-DSC	DKT-ST
ASSISTments2009	0.694	0.761	0.751	0.791	0.775	0.817
ASSISTments2017	--	0.723	0.654	0.754	--	0.767
KDDCup2010	--	--	--	--	0.723	0.739

IV. D. Analysis of knowledge tracking results

The goal of the knowledge tracking task also includes tracking the changes in the students' knowledge status, in order to observe the changes in the learners' knowledge level during the learning process, this section intercepts the exercise records of a learner in the ASSISTments2009 dataset, which contains the data of his interactions with the 21 exercises on the four knowledge points, and the output of the dataset's knowledge level results are shown in Fig. 3, in which the horizontal axis is the student's The horizontal axis is the sequence of answering questions, and the vertical axis indicates the knowledge points corresponding to the questions answered in that period of time. The change of ability attributes before and after learning is shown in Table 2, which shows the comparison of the ability attributes of a student before and after practicing 21 exercises in the educational scenario, and the radar chart can intuitively feel the enhancement of their own ability level after practicing, especially in the ability attribute numbered 0, which is enhanced by 0.46 after practicing 21 exercises.

In moments 1 and 2, students successively answered the exercises correctly, and the results of the model's tracking of the mastery level of Knowledge Point 2 improved significantly (the output value became higher, and the color of the color block became lighter); while in moment 15, the knowledge level of the students declined after they answered the exercises corresponding to Knowledge Point 2 incorrectly, which indicates that the model is able to effectively track the mastery level of the knowledge based on the students' historical performances, and modeling the learning process.

Observing the students from moment 3 to moment 12, the students did not answer the exercises corresponding to knowledge point 2, but their knowledge status has been changing with the interaction sequence of other knowledge points. It can be seen that the model proposed in this chapter is able to take into account students' individualized factors to a certain extent, conforms to the intuition of learners' knowledge mastery level in the educational process, and traces learners' mastery of each knowledge point under the change of time.

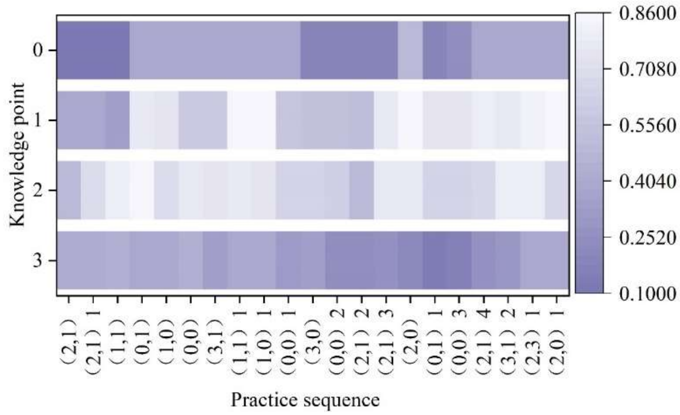

Figure 3: Data set knowledge level output

Table 2: Ability to change before and after learning

Ability number	T=0	T=21
0	0.22	0.68
1	0.65	0.75
2	0.58	0.64
3	0.46	0.59
4	0.39	0.61
5	0.48	0.77

In this section, the historical interactions of one learner's responses at 8 time steps in the ASSISTments2009 dataset are selected and the knowledge mastery level of the 2 knowledge points are outputted by the Deep Knowledge Tracking model and the DKT-ST model in this chapter, respectively.

The sequence of interactions for this student is: [(0,1),(0,1),(1,1),(0,1),(1,0), (1,1),(0,0),(1,0)]. Each of these items represents the student's answering behavior on the corresponding question, and the 2 sets of models track the level of knowledge acquisition as shown in Figure 4.

The student correctly answered the exercise question containing Knowledge Point 1 at moment 4, and both 2 groups of models reflected an increase in the level of knowledge at moment 4. It shows that both the DKT model and the DKT-ST model can achieve the purpose of modeling learning behavior and tracking the knowledge level. After the student incorrectly answered the question containing Knowledge Point 2 at the 5th moment, the change in knowledge level in the DKT output was relatively smaller, but the DKT-ST showed a continuous decrease, indicating that the DKT-ST takes into account other relevant characteristics of the learner's personalized learning process, constructs individual differences based on the learner's ability, and more accurately traces the changes in the knowledge mastery of the learner brought about by changes in the learner's ability.

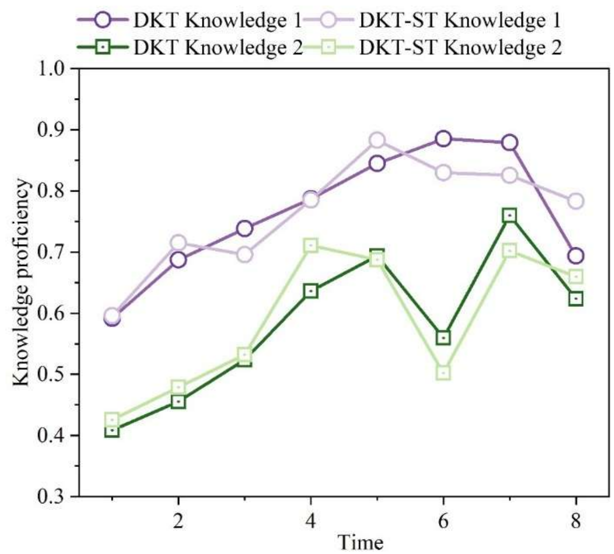

Figure 4: Comparison of knowledge state output

V. Analysis of the effects of improving students' capacity for independent learning

V. A. Experimental design

Pre-test experiment is the first step of this study, before the beginning of teaching need to understand the students' independent learning ability, in order to explore the impact of the teaching mode designed in this study on the students' independent learning ability, so before the beginning of teaching were issued to the experimental class and the control class at the same time to the students of the College Students' Independent Learning Ability Survey questionnaire, the two classes have 30 students each, the questionnaire was issued 60, the recovery rate of 60 copies of the questionnaire was 100%. 60 copies, the questionnaire recovery rate is 100%.

At the end of the whole semester's teaching, the questionnaire of “Survey on Independent Learning Ability of College Students” was distributed to the control class and the experimental class again, and 30 copies of the questionnaire were distributed to each class respectively, and there were 60 copies in total, with a 100% recovery rate of the questionnaire. The pre-test questionnaire was used to compare the data, so as to further analyze the impact of the teaching mode designed in this paper on students' independent learning ability, and how students' independent learning ability changed before, during and after learning.

V. B. Experimental Results of Self-directed Learning Ability Enhancement

The average scores of the pre-test of the experimental class and the control class are shown in Table 3, and by comparing the average values of the independent learning ability of the students in the experimental class and the control class, which are 69.86 and 69.99 respectively, it can be seen that there is not a big difference in the overall level of the experimental class and the control class.

Table 3: The average performance of the experiment class

	The previous test score of the experimental class	The total score of the comparison class
Mean value	69.85	69.99
Case	30	30
Standard deviation	10.15	11.03

The average post-test scores of the experimental and control classes are shown in Table 4, which shows that the average scores of both classes changed after the semester-long course. The overall level of both the experimental and control classes increased. The scores of the experimental class increased by 25.61, which is about 37.20% year-on-year compared to before the class. The scores of the control class also rose, with an increase of 10.42, which is about 14.89% year-on-year. This shows that the experimental class had a greater increase in independent learning ability.

Table 4: The average results of the experiment were measured

	The total score of the experimental class	The total score of the cross section
Mean value	95.46	80.41
Case	30	30
Standard deviation	7.56	7.49

The results of the before and after control of the level of autonomous learning in the experimental and control classes are shown in Table 5, where the students in the experimental group did not have strong abilities such as the ability of self-planning before receiving the teaching model of this paper. In the traditional teaching mode, although the students' self-learning ability changed compared to before, the average rise value of the experimental class was 0.99 and the average rise value of the control class was 0.34, and the rise of the experimental class was significantly higher than that of the control class.

Table 5: Independent learning level

		Mean	N	Standard deviation	Standard error mean
Laboratory class	Premeasured mean	2.99	30	0.854	0.145
	Backmeasured mean	3.98	30	0.856	0.149
Cross-reference class	Premeasured mean	3.12	30	0.946	0.165
	Backmeasured mean	3.46	30	0.895	0.153

VI. Conclusion

This study effectively solves the key technical problems in the cultivation of independent learning ability of English teacher training students by constructing a personalized learning model based on artificial intelligence algorithms. The DKT-ST algorithm excels in knowledge tracking performance, with an AUC value of 0.767 on the ASSISTments2017 dataset, which outperforms the traditional deep learning methods. The algorithm successfully introduces the forgetting mechanism and student individual difference modeling, which makes the knowledge state tracking more in line with the real learning process, and the accuracy of student ability assessment is significantly improved.

The teaching experiment verifies the practical application value of the technical program. The independent learning ability of students in the experimental class increased from an average of 69.85 to 95.46 points, and the improvement effect far exceeded the change from 69.99 to 80.41 points in the control class. The independent learning level of students in the experimental class increased from 2.99 to 3.98 points, with an increase of 0.99 points, while the control class only increased by 0.34 points, which fully proves the advantages of the intelligent teaching mode. The analysis shows that students' development in key competency dimensions is more balanced, and there is a significant improvement in motivation and ability to use learning strategies.

The open design of the ALS-OLM model provides a scalable architectural solution for the future development of educational technology, and its cross-platform data sharing and privacy protection mechanism has important promotion value. This technical solution not only enhances the independent learning ability of English teacher trainees, but also provides ideas for personalized teaching in other disciplines, which is of great significance in promoting the development of teacher education informatization.

Funding

This work was supported by:

1. Teaching Reform Research Project for Regular Undergraduate Universities in Hunan Province (Document No. XiangJiaoTong [2024] 147) - 'Research on the Cultivation of Teaching Practical Abilities for English Normal Students in Local Universities under the Background of Professional Certification' (Project Approval Number: 202401001401).
2. Chenzhou City Education Science '14th Five-Year Plan' 2024 Special Project on School-University Collaboration (Document No. ChenJiaoKeGui [2024] 1) - 'Research on the Blended Model of English Teacher Training Skill Development under the Framework of Teacher Education Professional Certification' (Project Approval Number: CJK24BXXHZ18).

References

[1] Delgado Hernández, Y. L., López Regalado, O., Calua Cueva, M. R., & Moreno Muro, J. P. (2021). AUTONOMOUS LEARNING IN THE EDUCATIONAL CONTEXT: A PSYCHOLOGICAL PERSPECTIVE. ASEAN Journal of Psychiatry, 22(9).

[2] Aminatun, D., & Oktaviani, L. (2019). Memrise: Promoting students’ autonomous learning skill through language learning application. Metathesis: Journal of English Language, Literature, and Teaching, 3(2), 214-223.

[3] Alrabai, F. (2017). From teacher dependency to learner independence: a study of Saudi learners’ readiness for autonomous learning of English as a Foreign Language. Learning and Teaching in Higher Education: Gulf Perspectives, 14(1), 70-97.

[4] Chen, L. (2024). Fostering independent learning in english education for undergraduate students. Journal of Social Science Development Research, 2(1), 36-39.

[5] Anggoro, K. J., & Khasanah, U. (2022). EFL Students' Independent Learning with Pear Deck Interactive Slides. Studies in Self-Access Learning Journal, 13(1).

[6] Soleymani, S., Aliabadi, K., Zaraii Zavaraki, I., & Delavar, A. (2021). The effect of flipped learning pattern (FLP) in terms of problem-based teaching approach (PBTA) on applicants’ self-directed learning (SDL) towards the English courses. Journal of Foreign Language Teaching and Translation Studies, 6(2), 87-116.

[7] Hawkins, M. W. (2018). Self-directed learning as related to learning strategies, self-regulation, and autonomy in an English language program: A local application with global implications. Studies in Second Language Learning and Teaching, 8(2), 445-469.

[8] Papamitsiou, Z., & Economides, A. A. (2019). Exploring autonomous learning capacity from a self‐regulated learning perspective using learning analytics. British Journal of Educational Technology, 50(6), 3138-3155.

[9] Tomakin, E. (2023). TERTIARY LEVEL STUDENTS'LEARNING ENGLISH: DEPENDENT INDEPENDENT LEARNERS. Asya Studies, 7(23), 27-39.

[10] Ovilia, R., & Ramadhani, S. (2023). EFL Students’ Readiness for independent Learning Observed from their Self-Regulation in Post COVID-19 Outbreak. AJELP: Asian Journal of English Language and Pedagogy, 11(2), 12-22.

[11] Mashitoh, N. L. D., Sukestiyarno, Y. L., & Wardono, W. (2021). Creative thinking ability based on self efficacy on an independent learning through google classroom support. Journal of Primary Education, 10(1), 79-88.

[12] Abdel Razeq, A. (2018). Autonomous learning levels of students majoring in EFL and the role of their teachers in developing autonomous learning. Journal of Educational and Psychological Studies-Sultan Qaboos University.

[13] Ramírez Espinosa, A., & Hernández Gaviria, F. (2024). Learning strategies instruction: A proposal for encouraging autonomous learners. Colombian Applied Linguistics Journal, 26(1), 78-90.

[14] Sutrisno, D., Inawati, I., & Hermanto, H. (2023). Enhancing student learning autonomously: Exploring the global impact of artificial intelligence. English Language Teaching Educational Journal, 6(2), 137-150.

[15] Okan Bulut,Jinnie Shin,Seyma N. Yildirim Erbasli,Guher Gorgun & Zachary A. Pardos. (2023). An Introduction to Bayesian Knowledge Tracing with pyBKT. Psych,5(3),770-786.

[16] Huanhuan Zhang,Lei Wang,Yuxian Qu,Wei Li & Qiaoyong Jiang. (2024). Enhanced Dynamic Key-Value Memory Networks for Personalized Student Modeling and Learning Ability Classification. Cognitive Computation,(prepublish),1-24.

[17] Feng Xu,Kang Chen,Maosheng Zhong,Lei Liu,Huizhu Liu,Xianzeng Luo & Lang Zheng. (2024). DKVMN&MRI: A new deep knowledge tracing model based on DKVMN incorporating multi-relational information.. PloS one,19(10),e0312022.