Learning Path Recommendation Based on Reinforcement Learning

Ji Li, Simiao Yu, Tiancheng Zhang

Abstract—**In recent years, online education platforms have seen rapid growth, attracting an increasing number of students to digital learning environments. In online education, learners can choose learning content and plan their own learning path more freely. Although the online education platform gives learners a high degree of freedom, it reduces the learning guidance for learners, which leads to problems such as "information overload" and "knowledge loss". The main manifestation is that learners don't know how to plan their learning path, resulting in reduced learning efficiency and poor learning effects. To address these challenges, this paper proposes a learning path recommendation algorithm based on reinforcement learning called RLLP. The RLLP model takes into account the learner's learning goals, knowledge level, and the relationships between knowledge points. Simultaneously, it also considers the smoothness of the learning path and the learner's engagement, aiming to recommend efficient and sensible learning paths to learners. Extensive experimental results demonstrate the effectiveness of RLLP model.**

Index Terms—**Online Education, Knowledge Tracking, Reinforcement Learning, Learning Path Recommendation, Proximal Policy Optimization**

I. INTRODUCTION

In recent years, online learning has successfully attracted a large number of learners, who register and utilize it due to its convenience, openness, and abundant learning resources. Learners have the opportunity to pursue independent learning based on their individual situations and preferences.

However, unlike traditional classroom settings, online education platforms often lack the capability to provide real-time supervision and guidance, which can lead to issues such as "information overload" and "knowledge loss". These problems are primarily manifested in the fact that learners struggle to plan their learning path amid an overwhelming array of resources of variable quality. Consequently, despite investing significant time into their studies, learners may struggle to effectively achieve their learning goals.

Intelligent Tutoring System (ITS), which provides personalized services for learners, has garnered widespread attention in the education sector since the 1970s.However, due to technological limitations during that era, ITS faced substantial challenges and failed to achieve significant breakthroughs. In recent years, with advancements in computer performance and artificial intelligence, ITS has undergone substantial development. ITS employs artificial intelligence technologies to allow computers to function as teachers within the educational environment, offering personalized assistance and guidance to learners based on their unique learning preferences and knowledge levels. One of the key research directions of ITS is the learning path recommendation.

One direction of learning path recommendation algorithm is based on complex network theory. Durand et al. [1] were the first to define the learning path as the topological sorting of several learning objects and proposed a learning path recommendation model based on graph theory, considering the dependency and order relationship between the learning objects. Zhu Y et al. [2] designed a knowledge model based on knowledge maps, taking into account factors such as learners' knowledge level and the correlation between courses to recommend suitable learning paths. Zhu H et al. [3] generated groups based on similarity to recommend learning paths. Zhu et al. [4] considered four different learning scenarios: initial learning, general review, pre-exam study, and pre-exam review, recommending learning paths according to the learning scenarios. Liu et al. [5] analyzed the learning relationship between courses and learners, adopting different learning path recommendation methods for different learners. Daqian et al. [6] constructed a multi-dimensional knowledge graph framework based on the semantic relationship between learners to recommend personalized learning paths. Tang et al. [7] generated learning paths for learners to learn from videos based on knowledge graphs.

Another direction of learning path recommendation algorithm is based on intelligence optimization algorithm. Wang et al. [8] first used the ant colony algorithm to recommend learning activities based on learners' learning styles. Kurilovas et al. [9] proposed a dynamic learning path selection method using an improved particle swarm optimization algorithm. Lin et al. [10] used an enhanced genetic algorithm with an approximate ideal ranking method to find the optimal learning path. Dwivedi et al. [11] considered learners' learning styles and knowledge levels, and recommended learning paths through variable-length genetic algorithms.

In addition to the methods mentioned above, there are many algorithms used to solve the learning path recommenddation problem. Fiqri et al. [12] used the improved Dijkstra algorithm to recommend learning paths

Manuscript received Oct 18, 2023; revised Jul 6, 2024.

Ji Li is a PhD candidate at the Northeastern University, Shenyang, China,110000. (e-mail: 408567077@qq.com).

Simiao Yu is a Lecturer at the University of Science and Technology Liaoning, Anshan, China,1140000.(corresponding author to provide. phone: 0412-5929096; fax: 0412-5929093; e-mail: 1115063992@qq.com).

Tiancheng Zhang is an associate professor at the Northeastern University, Shenyang, China,110000. (e-mail: tczhang@mail.neu.edu.cn).

for the purpose of shortening the learning time of learners. Xie et al. [13] divided the learners into groups, and planned the learning path according to the groups. Chungho et al. [14] considered the influence of learning style on learning effect, and generated learning paths of four learning styles for learners to choose. Wacharawan et al. [15] plan learning paths for learners from the perspective of context-aware computing. Zhu et al. [16] introduced a recurrent neural network to recommend learning paths for learners. Liu et al. [17] used reinforcement learning algorithms and considered the sequence relationship between exercises to recommend learning paths for learners.

In summary, this paper proposes a learning path recommendation algorithm called RLLP, based on reinforcement learning, designed to recommend learning paths for learners based on their learning goals. The specific contributions of this paper are as follows:

1) We build a learner simulator based on the knowledge tracing model. This simulator predicts learners' performance on the learning path and mastery of target knowledge points, utilizing static data from learners' educational records.
2) To ensure the rationality of our learning path recommendations, we consider the relationships between knowledge points. We design a knowledge point relationship mining algorithm based on association rules.
3) We design a learning path recommendation algorithm based on the Proximal Policy Optimization (PPO) algorithm, which considers the learners' learning goals, knowledge level, and the relationship between knowledge points, as well as the smoothness of the learning path and learners' participation, to recommend efficient and reasonable learning paths for learners.
4) We compared the RLLP model with advanced learning path recommendation algorithms on three real-world datasets. The experimental results demonstrated the effectiveness of the RLLP model in enhancing learning path recommendations.

II. Related Works

A. Knowledge Tracking Model

Knowledge tracking is a widely used technology in personalized guidance. Its task is to automatically track the changes in a learner's knowledge level based on their historical learning trajectory, to accurately predict their performance in future learning and provide corresponding assistance.

The knowledge tracking task can be formalized as: given a learner's historical learning interaction sequence $X_t = (x_1, x_2, \cdots, x_t)$ on a specific learning task, the objective is to predict the learner's performance on the subsequent interaction x_{t+1} .Each interaction x_t is characterized as (q_t, a_t) , where q_t denotes the question chosen by the learner at time t, and a_t represents the answering situation of the learner at time t. Knowledge tracing models can be roughly divided into those based on probabilistic graphical models, matrix factorization, and deep learning.

BKT (Bayesian Knowledge Tracing) [18] is one of the most used knowledge tracing models, which can evaluate learners' mastery of a certain knowledge point. The essence of BKT model is a hidden Markov model, its principle is to infer the unobservable state based on the observable state. BKT model represents a learner's knowledge state as a set of hidden variables, updating the probability distribution of these variables based on whether the learner can correctly answer questions. The BKT model represents learners' mastery of knowledge points through a set of binary variables, where each binary variable represents the learner's mastery of a specific knowledge point, 1 means mastered, and 0 means not mastered.

PMF (Probabilistic Matrix Factorization) was first applied in the field of recommendation. In recent years, researchers have improved the PMF algorithm and successfully applied it to knowledge tracking tasks. PMF model can calculate the degree of knowledge mastery of learners, rather than just judging whether a certain knowledge point is mastered. PMF model uses the learner's learning history matrix and the exercises-knowledge matrix. PMF model decomposes these matrices to obtain the latent feature representations of learners and exercises. These latent features can help understand learners' knowledge status and the characteristics of exercises. PMF model can predict learners' possible answer situations in the future, so as to realize personalized knowledge tracking and learning resource recommendation.

Deep learning, as a current research hotspot, has been widely applied in fields such as speech recognition, image classification, and natural language processing. In recent years, educational researchers have begun to use deep learning technology to solve difficult problems in the field of education. Piech et al. [19] proposed DKT (Deep Knowledge Tracing) model that uses deep learning to solve the problem of knowledge tracing. DKT model uses the temporal relationship through the recurrent neural network or the long-term short-term memory network to predict the next moment's performance based on the learner's historical learning records. DKT model first generates a one-hot vector through one-hot encoding of the learner's historical learning record, and inputs it into the LSTM network, extracts feature through the LSTM layer, then inputs the feature into the hidden layer, and finally outputs the prediction results from the output layer. The predicted results represent the learner's performance in the next question.

B. Reinforcement Learning

Reinforcement learning is an important branch of machine learning. Unlike supervised learning and unsupervised learning, reinforcement learning can learn autonomously through interaction with the environment. Owing to its robust performance in managing intricate decision-making problems that necessitate dynamic interaction and long-term strategizing, reinforcement learning has found extensive applications in fields such as robotic control [20] and game design [21].

The standard Reinforcement learning model includes four basic elements: environment, action, reward, and status. The interaction process between the agent and the environment can be summarized as follows: The agent chooses an action a_t in the current state S_t .The

environment calculates the state S_{t+1} of the agent at the next moment according to the action selected by the agent, and gives the agent a reward value r_t.The agent assesses the quality of its chosen action based on the reward value, and continues to select actions in the succeeding state, persisting in this process until termination condition is satisfied.

The types of reinforcement learning algorithms are as follows:

1) Value-based reinforcement learning algorithms

The value-based reinforcement learning algorithm implicitly constructs the optimal policy by obtaining the optimal value function, selecting the action with the highest value. Representative algorithms include Q-learning [22] and SARSA [23].

2) Policy-based reinforcement learning algorithms

Policy-based reinforcement learning algorithms do not require a value function, but directly search for the best policy and update the policy parameters by maximizing the cumulative reward.

3) Deep Learning based reinforcement learning algorithms

Traditional reinforcement learning algorithms have a limitation that each state and action is marked by a unique identifier. This leads to problems such as large storage space, long training time and poor training effectiveness when the state space is too large. The Deepmind team combined neural networks with Q-Learning algorithm to propose the Deep Q Network algorithm to solve the problem of Q-Learning algorithm requiring a lot of space and time.

III. Merhod

A. Symbol Definition

The symbols in this paper are given by Table I.

TABLE I
SYMBOL DEFINITION AND MEANING

Symbol Definition	Symbol Meaning
$S = \{S^1, S^2, \cdots, S^M\}$	learners' historical learning records
$S^i = \{S_1^i, S_2^i, \cdots, S_t^i\}$	historical learning records of learner i
$S_t^i = \{e_t^i, a_t^i\}$	exercises and performances chosen by learner i at time t
e_t^i	exercise chosen by learner i at time t
a_t^i	performance of learner i at time t
$L^i = \{e_1^i, \cdots, e_L^i\}$	learning path of learner i
$Q \in R^{M \times K}$	knowledge point matrix marked by experts
$V \in R^{M \times K}$	knowledge point matrix calculated by learner simulator
$U_i^t = \{U_{i1}^t, \cdots, U_{iK}^t\}$	mastery of knowledge points by learner i at time t
R_{ij}^t	Learner i's performance on exercise j at time t
$G \in R^{M \times N}$	learner score matrix

B. Problem Definition

The learning path recommendation involves arranging learning activities based on a learner's learning goals, study content, learning environment, and foundational knowledge under the guidance of specific learning strategies, with the aim of achieving learning goals. In traditional educational environments, the learning sequence of learners is arranged by teachers. In online educational environments, learners are required to independently plan their learning sequences. Therefore, learners frequently encounter challenges in devising suitable arrangements, even substantial time investment may not lead to the effective accomplishment of learning goals. The purpose of the learning path recommendation in this paper is to help learners arrange a reasonable learning sequence and help them better complete their learning goals.The learning path recommendation problem can be defined as:

Given a learner $i's$ learning records $S^i = \{S_1^i, S_2^i, \cdots, S_t^i\}$ and learning goal e_{target}^i, recommend Learning path $L^i = \{e_1^i, \cdots, e_L^i\}$, to maximize learner $i's$ mastery of learning goal e_{target}^i.

C. Model Overview

In this paper, we propose a learning path recommendation method based on the reinforcement learning PPO algorithm, referred to as RLLP. The structure of the RLLP model is shown in Figure 1.

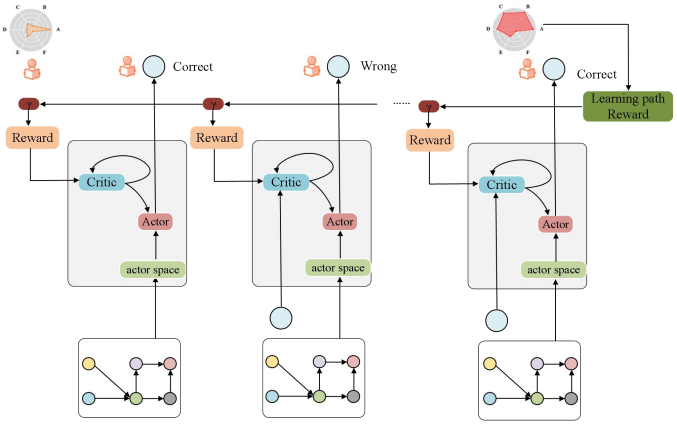

Fig. 1. Structure of RLLP model

The RLLP model consists of three parts: the learner simulator module, the knowledge points relationship mining module, and the reinforcement learning recommendation module. The student simulator module is used to simulate the learners' performance on the learning path and the change in knowledge point mastery. It judges the quality of the learning path and serves as a reward function for reinforcement learning. The knowledge points relationship mining module mines the relationship between knowledge points to enhance the rationality of the learning path and speed up the training speed of the reinforcement learning module. The reinforcement learning recommendation module recommends efficient and reasonable learning paths for learners based on the learner's learning goals, knowledge level, the relationship between knowledge points, the smoothness of the learning path, and the learner's participation. Next, we will introduce the details of the RLLP model.

D. Learner Simulator Module

To ensure the effectiveness of learning path recommendation, it is necessary to assess learners' performance on the recommended learning path and track the changes in their scores before and after learning. Learners' learning records are static, these data can't be directly observed. Therefore, the first step is to build the learner simulator to simulate learners using static data.

The role of the learner simulator is to evaluate learners' performance on the learning path and monitor changes in their mastery of knowledge points before and after learning. It helps assess the quality of the learning path and serves as the reward function for the reinforcement learning algorithm. In this paper, we employ the KPT model, as proposed by Chen [24], to build the learner simulator. The flow chart of the KPT model is shown in Figure 2.

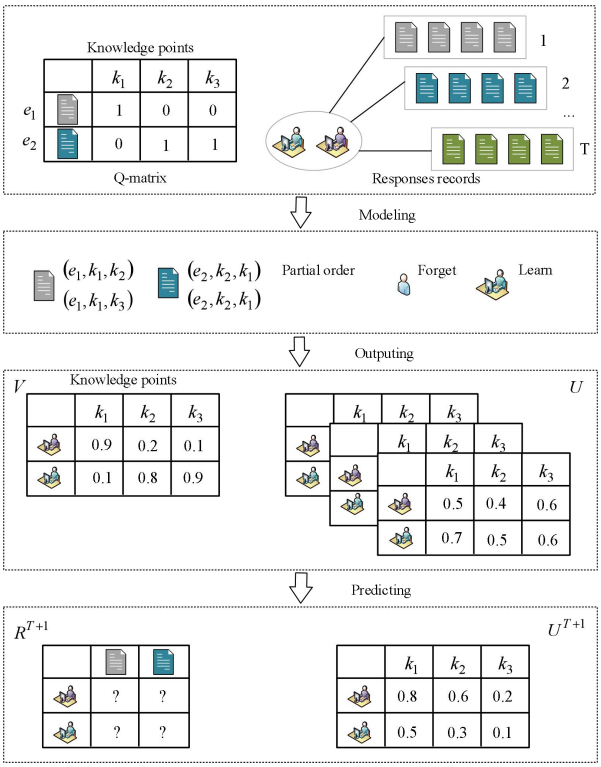

Fig. 2. Flow chart of KPT model

The KPT model is a sophisticated, interpretable probabilistic model that tracks learners' proficiency in various knowledge points through educational priors. The KPT model associates each exercise with a knowledge vector, where each element represents an explicit knowledge point. The Q matrix describing the relationship between the exercise and the knowledge point is marked by education experts. To track learners' knowledge proficiency, the relevant information of each learner is embedded into the same knowledge space. Importantly, the KPT model accommodates changes in knowledge proficiency over time by integrating the concepts of the memory curve and forgetting curve from traditional pedagogical theory into its modeling process.

The input of KPT model is the learner's performance matrix R_{ij}^{t} and the exercise knowledge matrix Q marked by experts. The output of KPT model is learner i's mastery of knowledge points at time t and the accurate excise knowledge point matrix V.

E. Knowledge Points Relationship Mining Module

In pedagogical theory, researchers generally believe that the learning sequence has a greater impact on the learning efficiency of learners, and this learning sequence is mostly related to the relationship between knowledge points. Therefore, when recommending learning paths to learners, the relationship between the learned knowledge points should be considered. Existing research classifies the relationship between knowledge points into the following three types:

1) Parent-child relationship. Parent-child relationship is a description of the overall and individual relationship between two knowledge points. Parent knowledge points are composite knowledge points composed of multiple basic knowledge points, while child knowledge points are the basic knowledge points that make up the parent knowledge point.
2) Sequence relationship. Sequential relationship refers to the order in which knowledge points are learned. For knowledge points A and B, if it is necessary to learn knowledge A before learning knowledge point B, then A is referred to as the predecessor knowledge point of B, and B is the successor knowledge point of A.
3) Parallel relationship. Parallel relationship means that there is neither parent-child relationship nor sequence relationship between knowledge points.

The knowledge points in this paper are all annotated by experts as fundamental knowledge points, so parent-child relationship is not considered in the mining of knowledge point relationships.

The knowledge point relationship graph is composed of knowledge points and edges describing the relationship between knowledge points, which plays a key role in providing learners with personalized learning guidance and evaluating learners' knowledge level. Next, we will introduce how to mine the sequence of knowledge points using association rule mining technology and generate knowledge point relationship graph.

Before mining association rules, digitize the learner's practice records into a score matrix G, as shown in (1).

$$G = \begin{bmatrix} g_{11}, g_{12}, \cdots, g_{1n} \\ g_{21}, g_{22}, \cdots, g_{2n} \\ \vdots, \quad \vdots, \quad \quad \vdots \\ g_{m1}, g_{m2}, \cdots, g_{mn} \end{bmatrix} \tag{1}$$

The rows represent the exercises, and the columns represent the learners. m is the total number of exercises. n is the total number of learners. When learner i completes exercise j correctly, the value of g_{ji} is 1, otherwise g_{ji} is 0. Next, we use the knowledge point matrix Q annotated by experts, the knowledge point matrix V calculated by the student simulator, and the score matrix G for association rule mining.

The first step is to calculate the consistency between two exercises. Consistency between exercises refers to the number of times two exercises are answered correctly or incorrectly by a learner simultaneously. The calculation formula is shown in (2).

$$Count(Q_a, Q_b) = \sum_{i=1}^{n} (g_{ai} \odot g_{bi}) \quad (2)$$

Where $\odot$ denotes the logical OR operation. The value of $g_{ai} \odot g_{bi}$ is 1 only when learner i answers both exercises correctly or both exercises incorrectly at the same time, otherwise the value is 0.

When $Count(Q_a, Q_b) < n \times 40\%$, it indicates that the relationship between the two exercises is weak. Therefore, the relationship between the two exercises will not be considered in the next steps.

The second step is to construct four types of exercise association rules:

1) When the learner answers exercise Q_a correctly, and answers exercise Q_b correctly at the same time.
2) When the learner answers exercise Q_b correctly, and answers exercise Q_a correctly at the same time.
3) When the learner answers exercise Q_a incorrectly, and answers exercise Q_b incorrectly at the same time.
4) When the learner answers exercise Q_b incorrectly, and answers exercise Q_a incorrectly at the same time.

These four exercise association rules can be summarized into two cases: from correct answer to correct answer and from wrong answer to wrong answer. The calculation formula is shown in (3).

$$Conf(Q_a \rightarrow Q_b) = \frac{Sup(Q_a, Q_b)}{Sup(Q_a)} \quad (3)$$

Where $Conf(Q_a \rightarrow Q_b)$ denotes the confidence of $Q_a \rightarrow Q_b$, $Sup(Q_a, Q_b)$ denotes the support between exercises, and $Sup(Q_a)$ denotes the support of exercise Q_a. When calculating the confidence of correct answers to correct answers, $Sup(Q_a)$ represents the number of times exercise Q_a is answered correctly, and $Sup(Q_a, Q_b)$ represents the number of times exercises Q_a and Q_b are simultaneously answered correctly. When calculating the confidence of wrong answers to wrong answers, $Sup(Q_a)$ represents the number of wrong answers to exercise Q_a, and $Sup(Q_a, Q_b)$ represents the number of wrong answers to both exercises Q_a and Q_b at the same time.

A higher value of confidence indicates a stronger association between the two exercises. Conversely, a lower value of confidence indicates a weaker association between the two exercises.

In order to eliminate unnecessary associations between exercises, a threshold value Min_{conf} is set for the confidence between exercises. When $Conf(Q_a \rightarrow Q_b) < Min_{conf}$, it is considered that there is no association between these two exercises.

The third step is to calculate the correlation between knowledge points. The calculation formula is shown in (4).

$$Rev(K_i, K_j)_{Q_a \rightarrow Q_b} = q_{ai} \times v_{bj} \times Conf(Q_a \rightarrow Q_b) \quad (4)$$

where K_i represents the knowledge points contained in exercise Q_a, and K_j represents the knowledge points contained in exercise Q_b. q_{ai} is given by the knowledge point matrix Q marked by experts. v_{bj} is given by the knowledge point matrix V calculated by the learner simulator.

The fourth step is to build a knowledge points relationship graph. In the previous steps, the association rules in two cases are considered, so there are two kinds of correlations between knowledge.

Assuming that the retained correlation $Rev(K_i, K_j)_{Q_a \rightarrow Q_b}$ is obtained by the association rule from the wrong answer to the wrong answer, the knowledge point K_i is the precursor knowledge point of the knowledge point K_j, and K_j is the successor knowledge point of the knowledge point K_i. Add a directed edge from knowledge point K_i to knowledge point K_j in the knowledge point relationship graph. The weight of the edge is $Rev(K_i, K_j)_{Q_a \rightarrow Q_b}$.

Assuming that the retained correlation $Rev(K_i, K_j)_{Q_a \rightarrow Q_b}$ is obtained by the association rule from the correct answer to the correct answer, based on the logical equivalence formula $Q_a \rightarrow Q_b = \sim Q_b \rightarrow \sim Q_a$,the knowledge point K_j is the precursor knowledge point of the knowledge point K_i, and K_i is the successor knowledge point of the knowledge point K_j. Add a directed edge from knowledge point K_j to knowledge point K_i in the knowledge point relationship graph. The weight of the edge is $Rev(K_i, K_j)_{Q_a \rightarrow Q_b}$.

If there are multiple edges between the knowledge point K_i and the knowledge point K_j, only the edge with the largest weight will be retained.

F. Reinforcement Learning Module

The objective of the reinforcement learning module proposed in this paper is to provide learners with personalized learning paths that maximize their learning goals according to the relationship between knowledge points and their mastery of knowledge points.

Reinforcement learning generally consists of four parts, namely Action, State, Reward, and Algorithm. The introduction will be explained in detail next.

1) Action

The action representation is shown in (5).

$$A_i = [a_1, ..., a_i, ..., a_M] \quad (5)$$

This action represents the exercise that the learner chooses next. Actions are represented by binary vectors whose length is the total number of exercises m. When the exercise i is selected, the value of a_i is 1, and the other parts are 0.

To ensure the rationality of the learning path, the reinforcement learning module is required to select actions based on the knowledge points relationship graph. The specific steps are as follows:

- Identify all predecessor knowledge points of the target knowledge point using the knowledge points relationship graph. Include exercises that contain these knowledge points in the relevant exercise pool.
- Locate the predecessor knowledge points of these identified predecessors and add exercises containing these newly identified knowledge points to the relevant exercise pool.
- Continue this process until no further predecessor knowledge points can be found. Any remaining exercises

are placed into the irrelevant exercise pool.

- When selecting an action, there is a 90% probability of choosing from the relevant exercise pool and a 10% probability of selecting from the irrelevant exercise pool.

2) State

The state representation of the Reinforcement Learning Module is shown in (6).

$$S = [u_1, u_2, ..., u_K, e_1, e_2, ..., e_M] \quad (6)$$

Where $u_1, u_2, ..., u_K$ represent the degree of mastery of knowledge points before learning, K is the number of knowledge points. $e_1, e_2, ..., e_M$ represent exercises, M is the number of exercises. When the learner chooses exercise i, the value of e_i is 1, and the other parts are 0.

3) Reward

The reward function used by the reinforcement learning module considers the change of learners' knowledge mastery and the two factors that affect the learning effect.

The change in the learners' mastery of the learning goals is shown in (7).

$$r_1 = \tilde{k}_{target} - k_{target} \quad (7)$$

Where, $\tilde{k}_{target}$ represents the learner's mastery of the learning goal after learning, k_{target} represents the learner's mastery of the learning goal before learning. $\tilde{k}_{target}$ and k_{target} are given by the learner simulator.

Research indicates that learning is a continuous process for learners, and the difficulty of exercises in the learning path should remain relatively consistent. Consequently, a smoothness factor is incorporated into the reward function to maintain this consistency. The calculation formula is shown in (8).

$$r_2 = -\sum_{t=1}^{L} (d_{t+1} - d_t)^2 \quad (8)$$

where d_{t+1} represents the difficulty of the next exercise, d_t represents the difficulty of the previous exercise, and L represents the length of the learning path. We hope that the difficulty between exercises is as close as possible, so r_2 is taken as negative.

Research in educational psychology indicates that learner engagement significantly affects learning efficiency. Two primary factors influence learner engagement:

- When problems are perceived as too easy, learners may find them insufficiently challenging, leading to reduced dedication to studying.
- Conversely, when problems are too difficult, learners may experience frustration or other negative emotions, which can lead to disengagement from the learning process.

Therefore, when recommending a learning path, it should be similar to the learner's mastery of the knowledge points, in order to ensure learners' participation. learners' participation. The calculation formula of the learner's participation is shown in (9).

$$r_3 = -\frac{1}{L}\sum_{t=1}^{L} (d_t - \varphi)^2 \quad (9)$$

Where φ is the difficulty threshold, which is the mean value of the learners' mastery of the knowledge points related to their learning goals.

Combining the above three factors, the reward function expression is shown in (10).

$$R = \tilde{k}_{target} - k_{target} - \alpha \sum_{t=1}^{L} (d_{t+1} - d_t)^2 - \frac{\beta}{L}\sum_{t=1}^{L} (d_t - \varphi)^2 \quad (10)$$

Where α and β are penalty parameters, and the value is 0-1. If you want to increase the influence of any factor, you can increase the corresponding parameter.

4) Algorithm

The reinforcement learning algorithm used in this paper is the Proximal Policy Optimization (PPO) algorithm, which is an improved algorithm based on the actor-critic algorithm. In the actor-critic algorithm, the actor decides the action based on the state of the environment, and the critic evaluates the potential reward value that can be obtained for this action. The actor and critic are represented by different neural networks, where the actor is a policy-based reinforcement learning model and the critic is a value-based reinforcement learning model. Although the actor-critic algorithm can learn autonomously in the environment, it often takes a long time to converge. This is because the training and sample efficiency of the actor-critic algorithm are often low. Learning simple strategies may require thousands of samples, while learning complex strategies may require even more. The PPO algorithm addresses this issue by restricting the update range of the actor network's policy using a truncation function. The structure diagram of the PPO algorithm is shown in Figure 3.

The actor network selects actions based on the current policy in the action space constructed based on the knowledge relationship graph. It updates the network parameters based on the dominance values provided by the critic, which are calculated as shown in (11).

$$L_{actor}(\theta) = E_t\left[min\left(r_t(\theta)A_t, clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)A_t\right)\right] \quad (11)$$

Where $r_t(\theta)$ indicates the ratio of the current moment to the previous moment of the strategy, is the hyperparameter used to limit the magnitude of the strategy update, taking a value of 0.1 or 0.2. The function $(\cdot)$ restricts the value of $r_t(\theta)$ to the interval $[1-\varepsilon, 1+\varepsilon]$.

The use of $clip(r_t(\theta), 1-\varepsilon, 1+\varepsilon)$ to limit the update range of the policy update not only reduces the complexity of the algorithm but also improves its flexibility and stability. A_t represents the gain that can be obtained under the current action, as given by the critic network.

In the PPO algorithm, the critic network is utilized to assess the quality of the selected action. The loss function of the critic network is shown in (12).

$$L_{critic}(\phi) = \sum_{t=1}^{T-1} (\gamma V_\phi(s_{t+1}) + R(a_t, s_t) - V_\phi(s_t)) \quad (12)$$

Where $R(a_t, s_t)$ represents the reward value that can be obtained by selecting action a_t in state s_t, as given by Equation (10). $V_\phi(s_{t+1})$ represents the expected return that can be obtained in the next state s_{t+1}. γ is a discount factor that attenuates future returns, encouraging the algorithm to prioritize accumulating returns in the current moment.

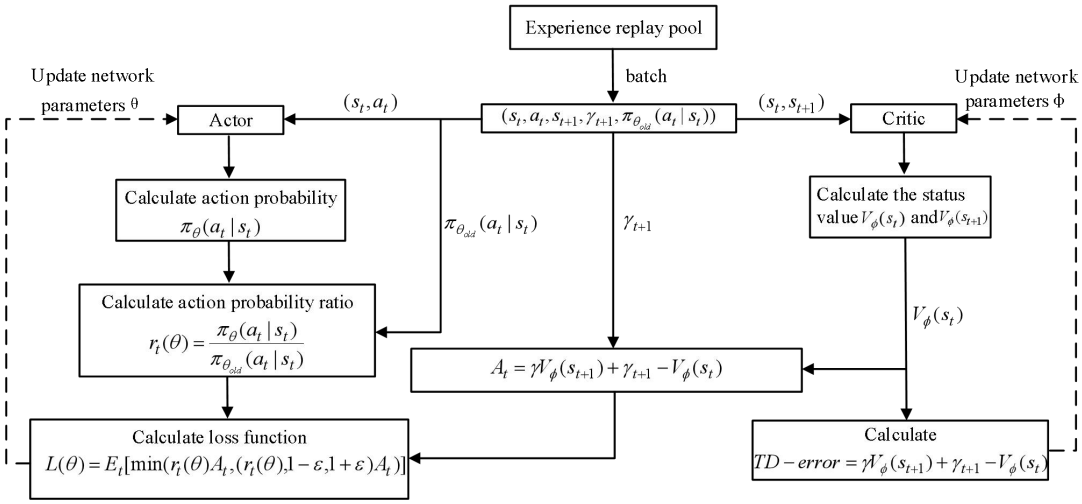

Fig. 3. Flow chart of PPO algorithm

IV. Experiment and Result

A. Datasets

The experiment in this paper used three public data related to mathematics. The statistical information of these datasets is shown in Table II.

TABLE II
Dataset description

dataset name	number of learners	number of exercises	number of knowledge points
FrcSub	536	20	8
Math1	4209	20	11
Math2	3911	20	16

B. Evaluation Indicators

Evaluation indicators for learning path recommendation algorithm primarily relies on the performance of learners before and after learning.

1) Effectiveness of learning

Learning effectiveness refers to the change in learners' scores before and after learning. The calculation formula is shown in (13).

$$E_p = \frac{P_{target\_e} - P_{target\_s}}{1 - P_{target\_s}} \tag{13}$$

Where P_{target_e} represents the probability that learners can answer exercises correctly containing the target knowledge points after learning, and P_{target_s} represents the probability that learners can answer exercises correctly containing the target knowledge points before learning. Based on different methods of calculating the probability of correctly answering exercises, we denote E_p as E_p_DKT and E_p_KPT. E_p_DKT represents the probability of correct exercise based on the DKT model, while E_p_KP represents the probability of correct exercise based on the KPT model.

2) Growth rate of mastery of target knowledge points

The calculation formula is shown in (14).

$$GR_{target} = \frac{K_{target_e} - K_{target_s}}{1 - K_{target_s}} \tag{14}$$

Where K_{target_e} represents the learners' mastery of the target knowledge points after learning, K_{target_s} represents the learner's mastery of the target knowledge points before learning.

3) Average correct rate on the learning path

The calculation formula is shown in (15).

$$ACR = \frac{1}{L}\sum_{i=1}^{L} P_{e_i} \tag{15}$$

Where L represents the length of the learning path, e_i represents the exercise i on the learning path, and P_{e_i} represents the probability of the learner doing the right exercise i, which is also calculated by the learner simulator.

C. Comparison Methods

1) KNN

K-Nearest Neighbor is a classic machine learning algorithm that recommends users based on the similarity between users.

2) GRU4Rec

GRU4Rec is a session-based recommendation method proposed in 2015. The input is the user's item record, and the output is the probability that the user selects the item at the next moment. The learning path recommendation algorithm based on GRU4Rec predicts the exercises that the learner may choose at the next moment based on the learner's practice record. It selects the exercises with the highest probability at each moment to form the learning path.

3) Q-Learning

This model utilizes the Q-Learning algorithm, using the skill mastery before and after learning as the reward function. However, it requires manually setting the state transition situations, and the Q-Learning algorithm can be time and space-consuming when dealing with large datasets.

4) DQN

This method is an improved algorithm based on Q-Learning. It utilizes the Deep Q Network algorithm to recommend learning paths for learners. By leveraging the powerful function approximation ability of neural networks, it addresses the issue of high time and space complexity in traditional Q-Learning algorithms.

D. Experimental Analysis

1) Comparison with other advanced methods

As shown in Figures 4-7, the RLLP model demonstrates significant improvements across various metrics when compared to other models. Specifically, in the dataset FrcSub, the RLLP model increased E_p_DKT by at least 6.0%, E_p_KPT by at least 4.6%, and GR_{target} by at least 4.9%. In the dataset Math1, the RLLP model increased E_p_DKT by at least 4.3%, E_p_KPT by at least 3.5%, and GR_{target} by at least 4.1%. Furthermore, in the dataset Math2, the RLLP model increased E_p_DKT by at least 4.7%, E_p_KPT by at least 5.3%, and GR_{target} by at least 5.7%. These results indicate that the learning paths recommended by the RLLP model significantly enhance learners' mastery of target knowledge points compared to other models.

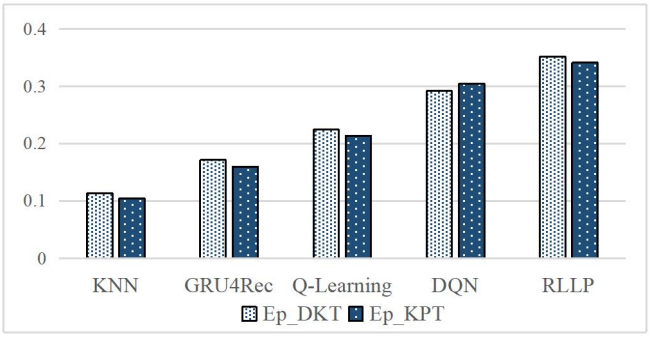

Fig. 4. E_p_DKT and E_p_KPT of the dataset FrcSub

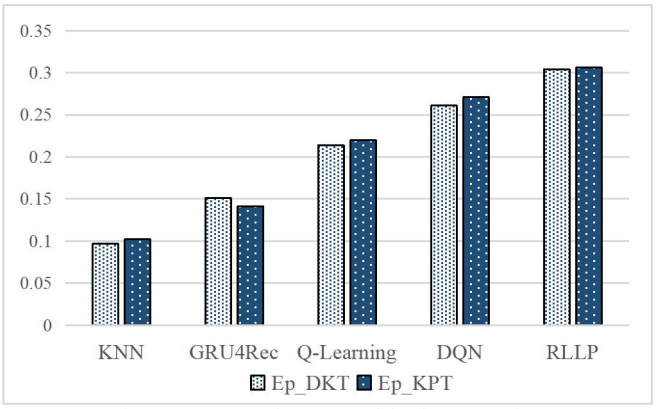

Fig. 5. E_p_DKT and E_p_KPT of the dataset Math1

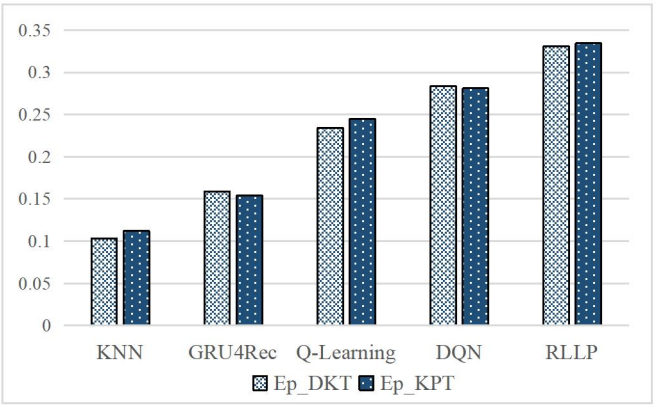

Fig. 6. E_p_DKT and E_p_KPT of the dataset Math2

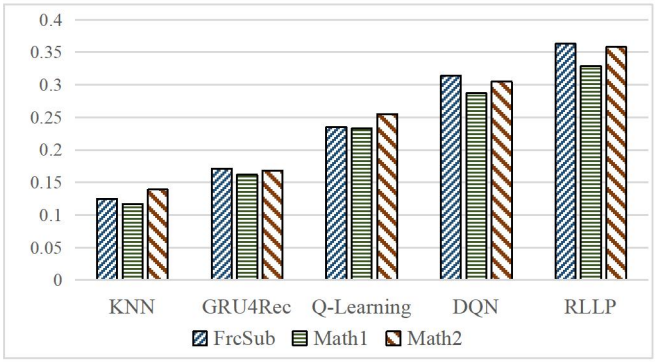

Fig. 7. GR_{target} of the data set FrcSub, Math1 and Math2

2) Correct rate on the recommended learning path

As shown in Figure 8, the KNN and GRU4Rec algorithms recommend exercises based solely on the characteristics of learners, without considering their specific learning goals. Consequently, the exercises recommended are often those already mastered by the learners, resulting in high correctness rates. However, these exercises contribute minimally to the advancement of learning goals. Compared with the other two reinforcement learning recommendation models, the RLLP model considers the smoothness of the learning path and the participation of learners, so the average correct rate will be higher.

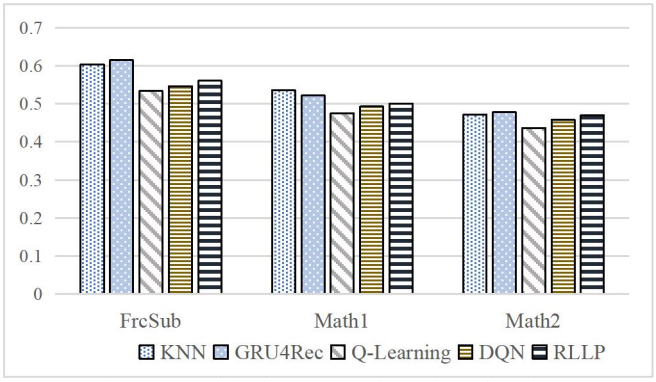

Fig. 8. *ACR* of the data set FrcSub, Math1 and Math2

3) The impact of the length of the learning path on the learning effect.

Figure 9-11 shows the GR_{target} of RLLP and four comparative experiments at different learning path lengths under the three datasets.

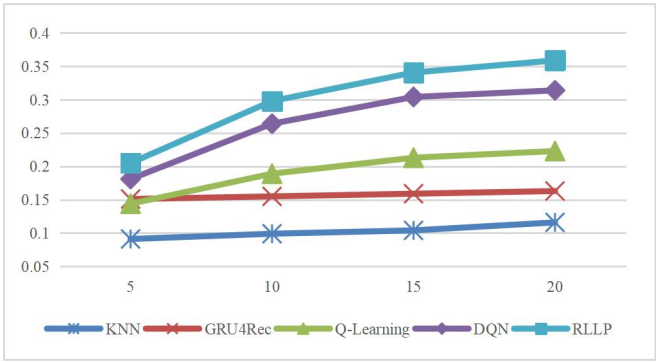

Fig. 9. GR_{target} of the dataset FrcSub with different learning path lengths

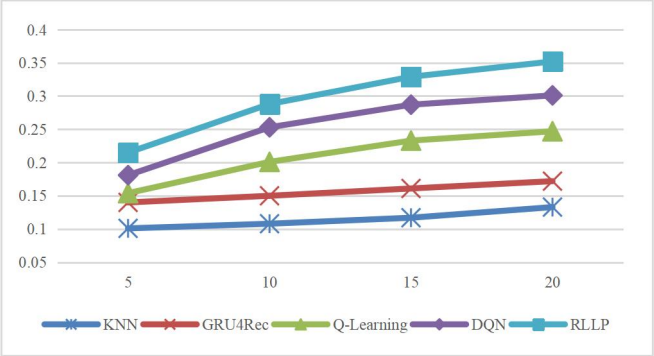

Fig.10. GR_{target} of the dataset Math1 with different learning path lengths

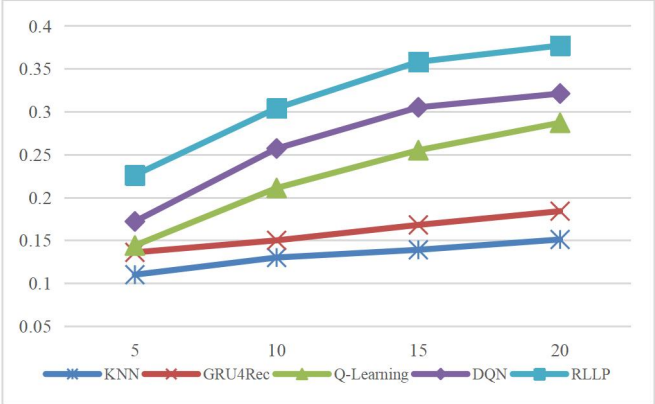

Fig.11. GR_{target} of the dataset Math2 with different learning path lengths

We can draw two conclusions:

- The RLLP model consistently outperforms other models across various learning path lengths, demonstrating its effectiveness.
- As the length of the learning path increases, there is a notable improvement in learners' outcomes. However, as the learning path becomes excessively long, the degree of improvement for the learners' learning goals gradually diminishes. Therefore, the length of the learning path should be appropriate.

V. Conclusion

In this paper, we propose a learning path recommendation method based on reinforcement learning and knowledge point relationship called RLLP. Initially, a learner simulator is constructed using the KPT model to emulate dynamic student behaviors from static data. Subsequently, we propose a knowledge point relationship mining algorithm that elucidates the relationships between knowledge points and generates a corresponding graph, thereby enhancing the rationality of the learning paths. Finally, we develop a reinforcement learning recommendation method based on the PPO algorithm. The RLLP model considers the learner's learning objectives, knowledge level, and the relationships between knowledge points. Simultaneously, it also considers the smoothness of the learning path and the learner's engagement, aiming to recommend efficient and sensible learning paths to the learners. Extensive experimental results demonstrate the effectiveness of RLLP model.

References

[1] Dwivedi S, Roshni V S K. Recommender system for big data in education[C].2017 5th National Conference on E-Learning & E-Learning Technologies (ELELTECH). IEEE, 2017: 1-4.

[2] Zhu Y, Wang P, Fan Y, Chen Y. Research of Learning path recommendation algorithm based on Knowledge Graph[C]. International Conference. ACM,2017:212-215.

[3] Zhu H, Liu Y, Tian F. A cross-curriculum video recommendation algorithm based on a video-associated knowledge map[J]. IEEE Access, 2018, 6: 57562-57571.

[4] Zhu H, Tian F, Wu K. A multi-constraint learning path recommend-dation algorithm based on knowledge map[J]. Knowledge-Based Systems, 2018, 143: 102-114.

[5] Liu H, Li X. Learning path combination recommendation based on the learning networks[J]. Soft Computing, 2020, 24(6): 4427-4439.

[6] Shi D, Wang T, Xing H, Xu H. A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning[J]. Knowledge-Based Systems, 2020, 195: 105618.

[7] Tang C L, Liao J, Wang H C. Supporting online video learning with concept map-based recommendation of learning path[C]. Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems. 2020: 1-8.

[8] Wang T I, Wang K T, Huang Y M. Using a style-based ant colony system for adaptive learning[J]. Expert Systems with Applications, 2008,34(4):2449-2464.

[9] Kurilovas E, Zilinskiene I, Dagiene V. Recommending suitable learning scenarios according to learners' preferences: An improved swarm-based approach[J]. Computers in Human Behavior, 2014, 30:550-557.

[10] Lin Y S, Chang Y C, Chu C P. An innovative approach to scheme learning map considering tradeoff multiple objectives[J]. Journal of Educational Technology & Society, 2016, 19(1): 142-157.

[11] Dwivedi P, Kant V, Bharadwaj K K. Learning path recommendation based on modified variable length genetic algorithm[J]. Education and information technologies, 2018, 23: 819-836.

[12] Fiqri M, Nurjanah D. Graph-based domain model for adaptive learning path recommendation[C]//2017 IEEE Global Engineering Education Conference (EDUCON). IEEE, 2017: 375-380.

[13] Xie H, Zou D, Wang F L. Discover learning path for group users: A profile-based approach[J]. Neurocomputing, 2017, 254: 59-70.

[14] Chungho S. Designing and developing a novel hybrid adaptive learning path recommendation system (ALPRS) for gamification mathematics geometry course[J]. Eurasia Journal of Mathematics, Science and Technology Education, 2017, 13(6): 2275-2298.

[15] Wacharawan I, Till B, Punnarumol T. Social Context-Aware Recom-mendation for Personalized Online Learning[J]. Wireless Personal Communications, 2017, 97 (1): 163-179.

[16] Zhu H, Tian F, Wu K, et al. A multi-constraint learning path recommendation algorithm based on knowledge map[J]. Knowledge-Based Systems, 2018, 143: 102-114.

[17] Liu Q, Tong S W, Liu C R, Zhao H K, Chen E H, Ma H P, Wang S J. Exploiting cognitive structure for adaptive learning[C]//Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2019: 627-635.

[18] Yudelson M V, Koedinger K R, Gordon G J. Individualized Bayesian Knowledge Tracing Models[C].16th International Conference on Artificial Intelligence in Education (AIED). Springer, Berlin, Heidelberg, 2013:171-180.

[19] Piech C, Spencer J, Huang J. Deep Knowledge Tracing [C]. Advances in Neural Information Processing Systems. 2015:505-513.

[20] Kober J, Bagnell J A, Peters J. Reinforcement learning in robotics: A survey[J]. International Journal of Robotics Research, 2013, 32(11): 1238-1274.

[21] Wang X, Sandholm T. Reinforcement learning to play an optimal Nash equilibrium in team Markov games[J]. Advances in Neural Information Processing Systems,2002, (15): 1571–1578.

[22] Watkins C, Dayan P. Q-learning[J]. Machine Learning,1992,8(3-4):279-292.

[23] Rummery G A, Niranjan M.On-Line Q-Learning Using Connectionist Systems [J].Technical Report,1994:112-118.

[24] Chen Y, Liu Q, Huang Z Y, Wu L, Chen E H, Wu R Z, Su Y, Hu G P. Tracking knowledge Proficiency of Students with Educational Priors[C]. The 26th ACM International Conference on Information and Knowledge Management (CIKM'2017), Singapore, November 6-10, 2017:989-998.

Ji Li is currently pursuing Ph.D. degree at Northeastern University in Shenyang, China. He obtained B.S. degree in Information and Computing Science from Shenyang University of Technology in 2018 and a M.S. degree in Computer Software and Theory from Northeast University in

Shenyang, China in 2021. His research direction is index and graph research..

Simiao Yu is a lecturer of of software engineering at the School of Computer and Software Engineering, University of Science and Technology Liaoning. Her main research interests include: big data technology, artificial intelligence technology, etc.

Tiancheng Zhang is an associate professor at the Institute of Computer Software and Theory, Northeastern University. He is also a member of ACM and China Computer Society. His main research interests include: big data technology, data flow analysis and mining, spatio-temporal data management technology, artificial intelligence technology, etc.