Digital Intelligence PRESS

Journal of Discovery Core

https://doi.org/10.67541/jdc2603

Research Article **Volume 1, Issue 1 (July 2026)** *Open Access*

Research on Personalized Art Learning Path Recommendation Algorithm Based on Reinforcement Learning

Sinuo Sun, Yansong Li*

Department of Art Studies, Harbin Conservatory of Music, Harbin, Heilongjiang, China

***Corresponding author:** Yansong Li. Email: liyansong813@163.com

Abstract

In online art education, the learning path has strong non-linear dependence and significant individual differences. Traditional sequential recommendation methods are inefficient because they struggle to dynamically model the skill topological constraints. This paper proposes a personalized art learning path recommendation algorithm named Art-DQN, which integrates graph attention networks and improved deep Q networks. Firstly, an art skill directed topological graph is constructed, and the graph attention network is used to extract high-order neighborhood features of the learner's skill state; Secondly, a dual-channel feature fusion module is designed, combining the skill graph features with the behavior sequence features encoded by LSTM and inputting them into the Q network; Finally, an ε-greedy exploration strategy based on skill topological constraints and a prioritized experience replay mechanism are proposed. Experiments on a dataset containing 500 real learners' logs and 1000 simulation trajectories show that the average skill mastery time of Art-DQN is 32.4 steps, which is 22.3% lower than that of the standard DQN, the target skill achievement rate reaches 92%, the number of prerequisite violations is 0, the path redundancy is as low as 0.09, and the convergence speed is increased by 41%. Ablation experiments verify the independent contributions of the dual-channel fusion and dynamic reward weights, and parameter sensitivity analysis and robustness tests further prove the stability of the algorithm. This algorithm provides an effective solution that combines structural constraints and personalized adaptation for intelligent recommendation in art education.

Keywords

Reinforcement learning; Personalized learning path recommendation; Graph attention network; Art education; Deep Q network

Received: 29 May 2026; Revised: 27 Jun 2026; Accepted: 11 Jul 2026; Published: 16 Jul 2026

1. INTRODUCTION

In recent years, with the development of digital technology and the popularity of online education platforms, art learning has gradually broken through the time and space limitations of traditional apprenticeship-style teaching [1]. More and more learners are obtaining art-related course resources such as painting, design, and music through the Internet. However, while online art education brings convenience, it also faces severe challenges in personalized recommendation. Unlike subjects with clear knowledge hierarchies and linear progression, art learning has strong nonlinear characteristics and individualized expression needs [2],[3],[4]. Learners' art foundation, aesthetic preferences, and creative styles vary, and the mastery speed and transfer effect of the same skill can vary significantly across learners. Existing recommendation systems mainly adopt collaborative filtering, knowledge graph-based path recommendation, or standard sequence recommendation methods [5],[6]. These methods

often assume that learning paths can be predefined by expert rules or statistically derived from group behavior patterns, which struggle to adapt to the common jumping progress, bottleneck periods, and cross-skill integration phenomena in art learning.

The contradiction between the nonlinear characteristics of art learning skill development and traditional sequence recommendation methods constitutes the core research motivation of this paper [7]. In traditional sequence recommendation, algorithms usually model the learning process as a fixed skill chain or tree-like structure, and the recommendation strategy tends to progress step by step according to the preset difficulty level [8],[9]. However, in art learning practice, the learning trajectory of learners is not a simple linear accumulation - a learner who already has strong observational ability may, through the practice of color courses, retroactively enhance their understanding of the light-dark relationship in sketching before fully mastering it. This non-monotonic, non-linear learning pattern makes the recommendation methods based on fixed sequence or a single knowledge graph rigid and inefficient [10],[11],[12]. Moreover, subjective factors such as learner interests and sudden inspiration further increase the difficulty of path prediction [13]. Therefore, designing a recommendation algorithm that can adapt to individual differences of learners, dynamically plan the learning sequence, and respect the inherent dependency structure of art skills becomes a key problem to be solved in the field of online art education.

To address these challenges, this paper proposes a path recommendation architecture that integrates an art skill topological graph with deep reinforcement learning. The main innovative contributions include the following three aspects. First, an art skill directed topological graph is constructed, and the graph attention network is used to structurally embed the learner's skill mastery status, enabling the state representation to explicitly model the pre-requisite dependencies and higher-order correlation information between skills, breaking through the limitation of traditional vector representation that cannot express skill relationships. Second, an improved deep Q-network with dual-channel feature fusion is designed, which integrates skill topological features and learning behavior temporal features in a unified Q-network, enabling the algorithm to not only perceive the global constraints of the knowledge structure but also capture the individual learner's behavior patterns and rhythm preferences. Third, an exploration strategy based on skill topological constraints and a priority experience replay mechanism are proposed, which enforce the pre-requisite rules in the reinforcement learning decision-making process, eliminate invalid recommendations, and accelerate algorithm convergence by prioritizing the sampling of high-difficulty skill transition samples. Through sufficient experiments on real art course logs and simulation extended data, the proposed algorithm has been verified to be significantly superior to existing baseline methods in terms of learning efficiency, path quality, and convergence performance, providing a technical solution with structural constraints and personalized adaptation capabilities for the intelligent recommendation scenario of art education.

2. ALGORITHM MODEL DESIGN

2.1 Modeling of skill dependency and graph embedding

Artistic learning exhibits distinct characteristics of skill progression. For instance, the perspective and light-dark relationships in sketching serve as the prerequisite for understanding light and shadow in color learning, and the ability to compose depends on the grasp of form and space [14],[15]. To formalize this dependency structure, this paper constructs a directed topological graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the node set $\mathcal{V} = \{v_1, v_2, \dots, v_N\}$ represents N atomic skill units (such as “line control”, “tone

transition", "color cold and warm matching", etc.), and the directed edge set $\mathcal{E} = \{(v_i \to v_j)\}$ represents that skill v_i is a necessary prerequisite for learning skill v_j. The construction of this graph is based on the expert knowledge in the field of art education and the hierarchical structure of the course outline, ensuring that there are no directed cycles in the graph, forming a topological structure of a directed acyclic graph [16].

For any learner, their skill mastery state at time t can be represented as a binary vector $\mathrm{m}_t \in \{0,1\}^N$, where $m_{t,i} = 1$ indicates that skill v_i has been mastered. However, merely having the mastery state is insufficient to describe the current learning position of the learner, because correlations exist between different skill nodes; mastering one skill may implicitly lead to partial mastery of its adjacent skills [17],[18],[19]. Therefore, this paper adopts graph attention networks to enhance the representation of the learner's skill state [20]. Given the node feature matrix $\mathrm{H} \in \mathbb{R}^{N \times d}$ corresponding to the currently mastered skill set, where each row h_i is the initial embedding vector of skill v_i (obtained by random initialization or pre-training), the graph attention mechanism calculates the attention coefficient of node v_i to its neighbor node $v_j \in \mathcal{N}(i)$ as follows:

$$\alpha_{ij} = \frac{\exp\left(\mathrm{LeakyReLU}\left(\mathrm{a}^{\top}[\mathrm{W}\mathrm{h}_i \parallel \mathrm{W}\mathrm{h}_j]\right)\right)}{\sum_{k \in \mathcal{N}(i)} \exp\left(\mathrm{LeakyReLU}(\mathrm{a}^{\top}[\mathrm{W}\mathrm{h}_i \parallel \mathrm{W}\mathrm{h}_k])\right)} \tag{1}$$

Here, $\mathrm{W} \in \mathbb{R}^{d' \times d}$ is a learnable linear transformation matrix, and $\mathrm{a} \in \mathbb{R}^{2d'}$ is the attention parameter vector. The $\parallel$ symbol represents the vector concatenation operation. $\mathcal{N}(i)$ is the set of neighbor nodes of node v_i (including the preceding and following skills). LeakyReLU serves as the non-linear activation function. After the single-layer graph attention, the updated representation of node v_i is:

$$\mathrm{h}'_i = \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}\, \mathrm{W}\mathrm{h}_j\right) \tag{2}$$

Here, $\sigma(\cdot)$ represents the ELU activation function. By stacking two layers of graph attention networks, the embedding vector of each skill node can aggregate the information of its higher-order neighborhood (i.e., the skill nodes at a distance of 2). The skill enhancement feature vector $\mathrm{g}_t \in \mathbb{R}^{d'}$ of the learner at time t is obtained by performing weighted pooling of the mastery states of all skills and the graph embeddings:

$$\mathrm{g}_t = \sum_{i=1}^{N} m_{t,i} \cdot \mathrm{h}'_i \tag{3}$$

The vector g_t not only contains the skills that the learner currently possesses, but also implicitly expresses through the graph structure the positions of these skills in the overall knowledge topology as well as information about skills that are adjacent to those mastered but not yet mastered themselves. This provides a structured state representation for the subsequent path recommendation [21].

2.2 Improved deep Q-network architecture (art-DQN)

Under the reinforcement learning framework, the path recommendation problem is modeled as a Markov decision process [22]. This paper proposes an improved deep Q-network architecture, called Art-DQN. The core innovation of this architecture lies in the composite design of the state space and

the dual-channel feature fusion network. The state space consists of two parts: the first part is the skill-enhanced feature vector $\mathrm{g}_t \in \mathbb{R}^{d'}$ at time t (from Section 2.1); the second part is the historical sequence of the learner's recent L learning behaviors. Let the behavior sequence be $\{a_{t-L+1}, a_{t-L+2}, \ldots, a_t\}$, where each action a_τ corresponds to a selected learning unit (belonging to the action space). Each action is mapped to a vector $\mathrm{e}(a_\tau) \in \mathbb{R}^{d_e}$ through a learnable embedding layer. Then, this sequence is input into a single-layer long short-term memory network (LSTM) to capture the temporal dependency patterns in the behaviors [23],[24]:

$$\mathrm{s}_t^{\mathrm{seq}} = \mathrm{LSTM}(\mathrm{e}(a_{t-L+1}), \mathrm{e}(a_{t-L+2}), \ldots, \mathrm{e}(a_t)) \tag{4}$$

Here, $\mathrm{s}_t^{\mathrm{seq}} \in \mathbb{R}^{d_s}$ represents the hidden state of the LSTM at the last time step. The final complete state vector is $\mathrm{s}_t = [\mathrm{g}_t \parallel \mathrm{s}_t^{\mathrm{seq}}] \in \mathbb{R}^{d'+d_s}$.

The action space $\mathcal{A}$ is the set of all candidate learning units, including specific skill practice items, micro-course modules, or creative tasks, totaling $|\mathcal{A}| = M$ discrete actions. The design of the reward function directly affects the quality of the learning path [25]. In this paper, the reward is decomposed into short-term and long-term components, and fused through dynamic weight coefficients. The short-term reward r_t^{short} is defined as the immediate efficiency of the learner after completing the current learning unit:

$$r_t^{\mathrm{short}} = \eta \cdot \mathbb{I}_{\mathrm{complete}} - \lambda \cdot \frac{T_t}{T_{\max}} \tag{5}$$

Here, $\mathbb{I}_{\mathrm{complete}}$ is an indicator function. It takes the value of 1 when the learner successfully completes the current learning unit and 0 otherwise. η represents the completion reward coefficient; T_t is the time spent on completing the current unit, and $T_{\max}$ is the expected maximum time consumption. λ is the time penalty coefficient. The long-term reward r_t^{long} measures the progress in skill acquisition and learning retention after this learning session:

$$r_t^{\mathrm{long}} = \sum_{i=1}^{N} \left(\tilde{m}_{t+1,i} - \tilde{m}_{t,i}\right) \cdot w_i + \beta \cdot \mathrm{Retention}(t) \tag{6}$$

Here, $\tilde{m}_{t,i} \in [0,1]$ represents the continuous level of skill mastery (distinct from the binary version), and w_i is the importance weight of skill v_i (determined based on its out-degree in the topology graph or expert annotations); $\mathrm{Retention}(t)$ is the learning retention function, which can be set as the normalized value of the learner's test accuracy for the current unit content one week later, and β is its weight coefficient. The overall reward is:

$$r_t = \gamma_t \cdot r_t^{\mathrm{short}} + (1 - \gamma_t) \cdot r_t^{\mathrm{long}} \tag{7}$$

Here, $\gamma_t \in [0,1]$ is the dynamic weight adjustment factor, defined as $\gamma_t = \sigma(\theta \cdot (t_{\mathrm{current}} - t_{\mathrm{mid}}))$; that is, in the early learning stage, it leans towards long-term rewards to establish a solid foundation, and gradually increases the weight of short-term rewards in the later learning stage to improve efficiency.

The innovation of the network structure of Art-DQN lies in the dual-channel feature fusion module. The skill-enhanced feature g_t and the behavioral sequence feature $\mathrm{s}_t^{\mathrm{seq}}$ are respectively passed through two independent fully connected encoding layers, resulting in dimensionally aligned intermediate representations $\mathrm{f}_t^{\mathrm{skill}}$ and $\mathrm{f}_t^{\mathrm{behav}}$. Subsequently, they are interacted through a fusion layer:

$$\mathrm{f}_t^{\text{fused}} = \mathrm{ReLU}\left(\mathrm{U}\mathrm{f}_t^{\text{skill}} + \mathrm{V}\mathrm{f}_t^{\text{behav}} + \mathrm{b}\right) \tag{8}$$

Here, U, V are learnable fusion matrices, and b is the bias term. The fused features are then passed through two fully connected layers to output the Q value $Q(\mathrm{s}_t, a; \theta)$ for each action, where θ represents all the parameters of the network. The update of the Q value adopts the standard temporal difference loss function of DQN [26]:

$$\mathcal{L}(\theta) = \mathbb{E}_{(\mathrm{s}_t, a_t, r_t, \mathrm{s}_{t+1}) \sim \mathcal{D}} \left[\left(r_t + \max_{a'} Q(\mathrm{s}_{t+1}, a'; \theta^-) - Q(\mathrm{s}_t, a_t; \theta) \right)^2 \right] \tag{9}$$

Here, $\mathcal{D}$ represents the experience replay pool, and θ^- is the parameter of the target network. Every fixed number of steps, the parameters are updated by copying from θ.

2.3 Path stability and exploration strategy improvement

The ε-greedy strategy in standard DQN explores the action space in a uniformly random manner, which may cause serious problems in the art learning scenario: the learner may be recommended higher-level content for which they have not yet mastered all the prerequisite skills, thereby causing learning frustration and resulting in path interruption. To address this, this paper proposes an ε-greedy strategy based on skill topological constraints. Let $\mathcal{V}_{\text{mastered}} \subseteq \mathcal{V}$ be the set of skills mastered by the current learner. For any candidate action a corresponding to the skill unit v_a, its pre-requisite set is denoted as $\mathrm{Pre}(v_a) = \{v_i \in \mathcal{V} : (v_i \to v_a) \in \mathcal{E}\}$. The feasibility flag of the action is defined as:

$$\mathrm{Feasible}(a) = \begin{cases} 1 & \text{if } \mathrm{Pre}(v_a) \subseteq \mathcal{V}_{\text{mastered}} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

The improved exploration strategy conducts exploration with probability ε, but the explored actions are only uniformly sampled from the feasible action set $\mathcal{A}_{\text{feasible}} = \{a \in \mathcal{A} : \mathrm{Feasible}(a) = 1\}$; it proceeds with exploitation with probability $1 - \varepsilon$, selecting the action with the highest current Q value, which is also constrained within the feasible action set. This constraint ensures that the recommended learning units always satisfy the prerequisite dependency of skills throughout both the exploration and exploitation phases, fundamentally avoiding ineffective recommendations.

The sampling mechanism of the experience replay pool has also been specifically improved. The traditional uniform sampling treats all experience transfer samples equally, but the difficulty of different skill transitions varies significantly [27],[28]. The skill transition difficulty index $\mathrm{Diff}(v_i \to v_j)$ is defined as the normalized value of the average number of learning units required to master skill v_j (counted from mastering v_i). For an experience sample $(\mathrm{s}_t, a_t, r_t, \mathrm{s}_{t+1})$, its priority p is defined as:

$$p = \left(\max_{v_j \in \Delta\mathcal{V}} \mathrm{Diff}(v_{\text{prev}} \to v_j) \right)^{\alpha} \tag{11}$$

Here, $\Delta\mathcal{V}$ represents the set of newly acquired skills from s_t to s_{t+1}, while v_{prev} denotes the skill recently mastered before the transition. The parameter $\alpha \geq 0$ is an exponential parameter used to adjust the priority intensity. The sampling probability is $P(i) = \frac{p_i^{\beta}}{\sum_k p_k^{\beta}}$, where β is the normalization index parameter. This prioritized sampling mechanism enables more frequent replay of learning experiences involving high-difficulty skill transitions or crossing multiple levels of skills, thereby accelerating the algorithm's learning of the critical path. Additionally, to compensate for sampling bias,

an importance sampling weight $w_i = \left(\frac{1}{|\mathcal{D}|} \cdot \frac{1}{P(i)}\right)^{\nu}$ is introduced when calculating the temporal difference loss, where ν is the deviation correction coefficient. Through these two improvements, Art-DQN not only ensures the validity of the learning path topology but also significantly enhances the utilization efficiency of critical learning experiences, making the recommended path both stable and efficient.

3.EXPERIMENTAL DESIGN AND VALIDATION

3.1 Experimental environment and dataset

In order to evaluate the performance of the proposed algorithm under controllable and reproducible conditions, this paper builds a simulation environment specifically for the art learning scenario based on the OpenAI Gym framework, called ArtLearnEnv [29]. The core of this environment is a skill transfer probability model and a knowledge decay model based on the Ebbinghaus forgetting curve [30]. There are $N = 48$ atomic skill nodes in the skill topology graph, which are divided into three levels: basic (sketching, lines), advanced (color, texture), and high-level (composition, creative style). The actual mastery degree $p_{t,i} \in [0,1]$ of skill v_i by the learner at time t follows the following dynamic update rule: When the learner is recommended to learn unit a (corresponding to skill v_a) and completes the learning, the update of their mastery degree is:

$$p_{t+1,i} = p_{t,i} + \xi_{a,i} \cdot \left(1 - p_{t,i}\right) \cdot \exp\left(-\frac{|\, d(i,a)\,|}{\tau}\right) \tag{12}$$

Here, $\xi_{a,i} \sim \text{Beta}(\alpha = 2, \beta = 5)$ represents the random transfer gain coefficient, $d(i,a)$ is the shortest path distance between skill v_i and the target skill v_a in the topology graph, and $\tau = 2$ is the distance attenuation constant. This formula indicates that learning a skill not only enhances the skill itself but also promotes the mastery of adjacent skills in an attenuated manner. The forgetting model attenuates the skills that have not been reviewed every $\Delta t = 3$ simulation days:

$$p_{t+\Delta t,i} = p_{t,i} \cdot \exp\left(-\frac{\Delta t}{\kappa_i}\right) \tag{13}$$

Among them, κ_i represents the half-life parameter of skill v_i, with a value range of 5 to 20 days (the half-life of basic skills is longer). The simulation environment also includes a fatigue factor $f_t \in [0,1]$ for learners, which affects the probability of completing the learning: $P_{\text{complete}} = \sigma(\rho \cdot (p_{t,\text{current}} - f_t))$, where $\rho = 3.0$. f_t increases linearly with consecutive learning steps and resets after a rest.

The dataset consists of two parts. The first part is the real-world collected behavioral logs of art courses, sourced from the learning records of an online art education platform over a 3-month period, and 500 learners with more than 20 interactions were selected. Each log contains the learner's ID, timestamp, learning unit identifier, completion status (success/abandonment/partially completed), and self-assessed skill level before and after the lesson (a 5-level Likert scale). 15,842 valid learning transfer samples were extracted from these original logs. The second part is synthetic extended data, generated using the ArtLearnEnv simulation environment to align with the distribution of the real data, with 1000 complete learning trajectories (each trajectory contains 30 to 60 learning steps) to compensate for the lack of long-tail skill samples in the real data [31]. The final merged dataset is divided into training set, validation set, and test set in a ratio of 7:1.5:1.5, where the training set is used for Art-DQN and all

baseline models' parameter learning, the validation set is used for hyperparameter tuning, and the test set is used for final performance evaluation [32].

A total of four baseline groups were set for comparison. Random recommendation selects a learning unit uniformly at random from the set of actions at each decision step. Knowledge Graph-based path recommendation (KG-Rec) uses the same skill topology graph as this paper, adopts a breadth-first search rule, and prioritizes recommending the unlearned predecessor nodes closest to the current skill mastery center. Standard DQN uses traditional state representation (only the skill mastery vector) and uniform experience replay. The PPO algorithm, as a representative baseline of advanced policy optimization methods, uses the same state and action spaces, and sets the clipping parameter of its target function to 0.2. All reinforcement learning algorithms are trained for 2000 rounds, with a maximum step limit of 50 steps per round (if all target skills are achieved before, the training is terminated early). The learning rate is uniformly set to 3×10^{-4}, the discount factor $\gamma = 0.95$, and the target network update frequency is 100 steps [33].

3.2 Evaluation indicators

This paper sets evaluation indicators from three dimensions: learning efficiency, path quality, and algorithm performance. In terms of learning efficiency, the average skill acquisition time T_{master} is defined as the average number of learning steps required for learners to master all 48 skill nodes from the initial state (calculated based on all trajectories in the test set), and the fewer the steps, the higher the efficiency. The target skill achievement rate R_{goal} examines the proportion of learners who complete the preset 5 advanced target skills (corresponding to creation, color composition, spatial perspective, etc.) within the given maximum step limit (set at 40 steps):

$$R_{\text{goal}} = \frac{1}{K}\sum_{k=1}^{K} \mathbb{I}\left(\max_{t \le T_{\max}} p_{t,\text{goal}_k} \ge 0.8 \right) \tag{14}$$

Here, $K = 5$ represents the total number of target skills, $T_{\max} = 40$ is the maximum allowable number of steps, and $\mathbb{I}(\cdot)$ is the indicator function.

In terms of path quality, the number of prerequisite violations is defined as the total count of learning units that do not satisfy the prerequisite relationship in the recommended sequence. This indicator directly reflects the degree to which the algorithm complies with the knowledge topological constraints, and the ideal value is 0. The "Redundancy" of the learning path is used to measure the proportion of unnecessary learning in the path:

$$\text{Redundancy} = \frac{1}{L_{\text{path}}}\sum_{t=1}^{L_{\text{path}}} \mathbb{I}\left(\max_{j \in \text{Succ}(a_t)} \text{Dist}(a_t,\text{Goal}) > \text{Dist}(a_{t-1},\text{Goal}) \right) \tag{15}$$

Here, L_{path} represents the total number of steps in the path, $\text{Succ}(a_t)$ is the set of subsequent skills for action a_t, and $\text{Dist}(a,\text{Goal})$ is the shortest path length from the skill node a to any target skill. Redundancy is the proportion of steps in the path that cause the distance from the goal to be greater.

In terms of algorithm performance, the convergence speed is measured by the training round number E_{conv}, defined as the minimum number of rounds required for the average reward over 50 consecutive rounds to reach 95% of the final convergence value. The average reward convergence value Q_{final} is the average single-step reward over the last 100 training rounds. The reward function is as

described in Section 2.2, where the coefficients are set as: $\eta = 5.0$, $\lambda = 0.5$, $\beta = 0.3$, and the initial dynamic weight $\gamma_0 = 0.2$, the final value $\gamma_\infty = 0.7$.

3.3 Analysis of experimental results

First, a complete evaluation of all five methods was conducted on the test set, and the results are shown in Table 1. Art-DQN significantly outperformed all the baseline methods in all core indicators. In terms of the average time for mastering skills, Art-DQN only needed 32.4 steps to complete the learning of all 48 skills, which was 22.3% less than that of the standard DQN (41.7 steps) and 16.7% less than that of PPO (38.9 steps). In terms of the achievement rate of target skills, Art-DQN reached 92%, while random recommendations were only 44%. Particularly noteworthy is the number of violations due to pre-requisite violations. Random recommendations and standard DQN, due to the lack of topological constraints, respectively generated 5.8 and 3.2 violations on average, while Art-DQN achieved zero violations, fully verifying the effectiveness of the skill topological constraint ε-greedy strategy. The path redundancy of Art-DQN was 0.09, which was much lower than 0.21 of KG-Rec, indicating that its path was more compact and efficient.

Table 1. Overall performance comparison of each algorithm on the test set

Algorithm	Average skill acquisition time (steps) ↓	Target skill achievement rate (%) ↑	Number of violations of preposition usage ↓	Path redundancy ↓	Average reward convergence value ↑
Random	58.3 ± 4.2	44	5.8 ± 1.1	0.34 ± 0.05	1.82 ± 0.21
KG-Rec	46.7 ± 3.1	68	0.0 ± 0.0	0.21 ± 0.03	2.45 ± 0.18
Standard DQN	41.7 ± 2.8	76	3.2 ± 0.9	0.18 ± 0.04	3.12 ± 0.24
PPO	38.9 ± 2.5	81	2.4 ± 0.7	0.14 ± 0.03	3.56 ± 0.20
Art-DQN	32.4 ± 2.0	92	0.0 ± 0.0	0.09 ± 0.02	4.38 ± 0.17

Figure 1 presents the average reward convergence curves of five algorithms during training. The x-axis represents the number of training rounds (0 - 2000), and the y-axis represents the average reward value every 100 rounds (the average of 10 independent runs was taken and smoothed using a moving average with a window size of 5).

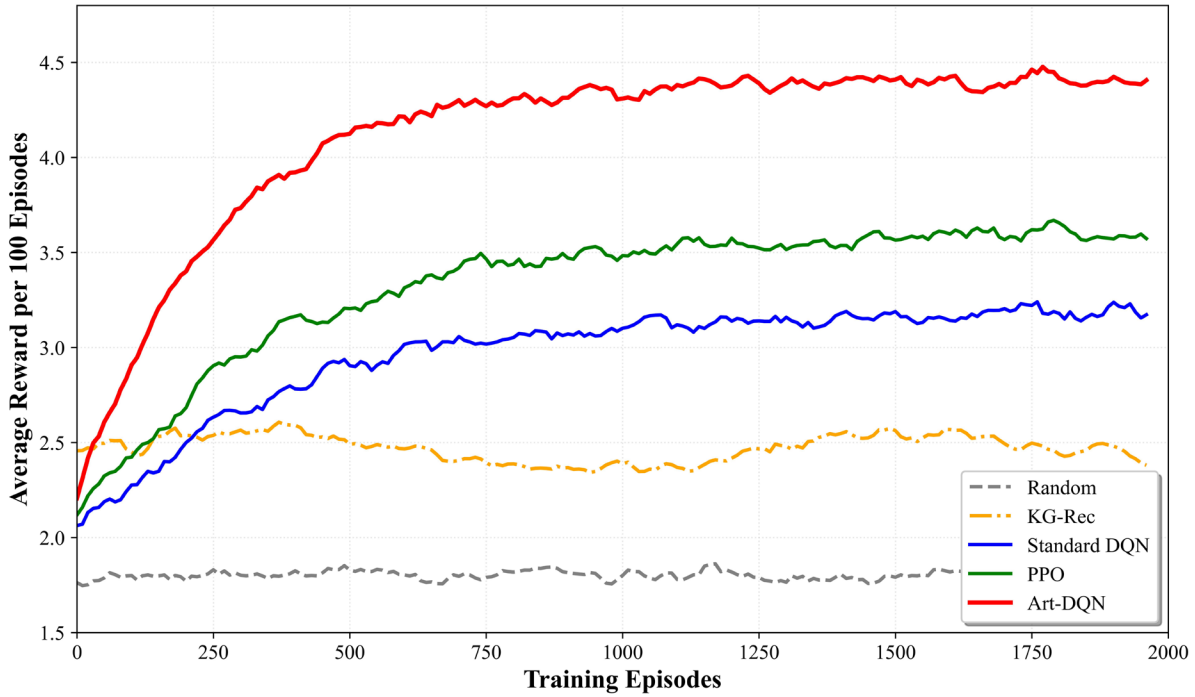

Figure 1. Convergence curves of training rewards for different algorithms in the ArtLearnEnv environment

Random recommendation and KG-Rec, due to their lack of learning ability or weak learning ability, always maintain a relatively low reward curve (converging to around 1.8 and 2.5 respectively). The standard DQN experiences a rapid increase in the first 300 rounds, but then levels off after approximately 800 rounds, eventually converging to around 3.1. PPO converges slightly faster than the standard DQN, with a final value of approximately 3.6. The curve of Art-DQN shows the steepest initial upward slope, reaching 95% of the convergence value (convergence round number $E_{\text{conv}} = 460$) around the 450th round, while PPO requires 620 rounds and the standard DQN requires 780 rounds. The final convergence value of Art-DQN reaches 4.38, significantly higher than other methods, indicating that the dual-channel feature fusion makes the Q-value estimation more accurate, and the prioritized experience replay accelerates the learning of high-value transfer patterns.

To quantify the independent contributions of the two core innovations, namely the dual-channel feature fusion module and the dynamic reward weight, two ablation variants were designed: Art-DQN w/o Fusion, where the dual-channel fusion is replaced with simple vector concatenation and directly input to the fully connected layer (retaining all other settings); Art-DQN w/o Dynamic Weight, where the dynamic weight γ_t in the reward function is fixed to a constant 0.5 (i.e., short-term and long-term rewards are equally weighted). Table 2 shows the performance comparison of the three versions on the test set.

Table 2. Comparison of ablation experiment results

Model variant	Average skill acquisition time (steps) ↓	Target skill achievement rate (%) ↑	Path redundancy ↓	Average reward convergence value ↑
Art-DQN (complete)	32.4	92	0.09	4.38
Art-DQN w/o Fusion	36.8 (+13.6%)	84	0.13	3.91
Art-DQN w/o Dynamic Weight	35.1 (+8.3%)	87	0.11	4.05

After removing the dual-channel fusion module, the average skill acquisition time increased from 32.4 steps to 36.8 steps, with a relative increase of 13.6%, while the target skill achievement rate decreased by 8 percentage points. This indicates that the interactive fusion of skill graph features and behavioral temporal sequence features is crucial for the completeness of state representation: simple concatenation cannot fully exploit the nonlinear correlation between the two types of features, resulting in underfitting of Q-networks for certain skill transfer patterns (such as multi-level learning transitions) that require long-term memory. The impact of removing the dynamic reward weight was relatively small but still significant, with the average skill acquisition time increasing by 8.3% and the reward convergence value dropping to 4.05. Fixed weights made the algorithm overly focus on short-term efficiency in the early exploration stage and neglect the solid mastery of basic skills, and lacked sufficient short-term rewards to optimize the learning rhythm in the later stage, leading to a slight increase in the final path redundancy.

Art-DQN involves two key hyperparameters: exploration rate decay coefficient $\varepsilon_{\text{decay}}$ and experience replay priority coefficient α. Figure 2 shows the average number of rounds required to converge (on the x-axis is the decay coefficient, on the y-axis is the convergence rounds) under different $\varepsilon_{\text{decay}}$ settings (0.90, 0.95, 0.99, 0.995, 0.999), as well as the corresponding final average reward values (represented by a line superimposition, on the right y-axis). When $\varepsilon_{\text{decay}} = 0.99$, the convergence speed is the fastest (460 rounds), and the final reward is the highest (4.38). A too small decay coefficient (0.90) leads to premature exploration termination, causing the algorithm to fall into local optimum (the final reward is only 3.65); a too large decay coefficient (0.999) prolongs the exploration stage, delays convergence to 720 rounds, but due to more thorough exploration, the final reward still remains at a relatively high level (4.29). This result indicates that the decay coefficient of 0.99 achieves the best balance between convergence speed and final performance.

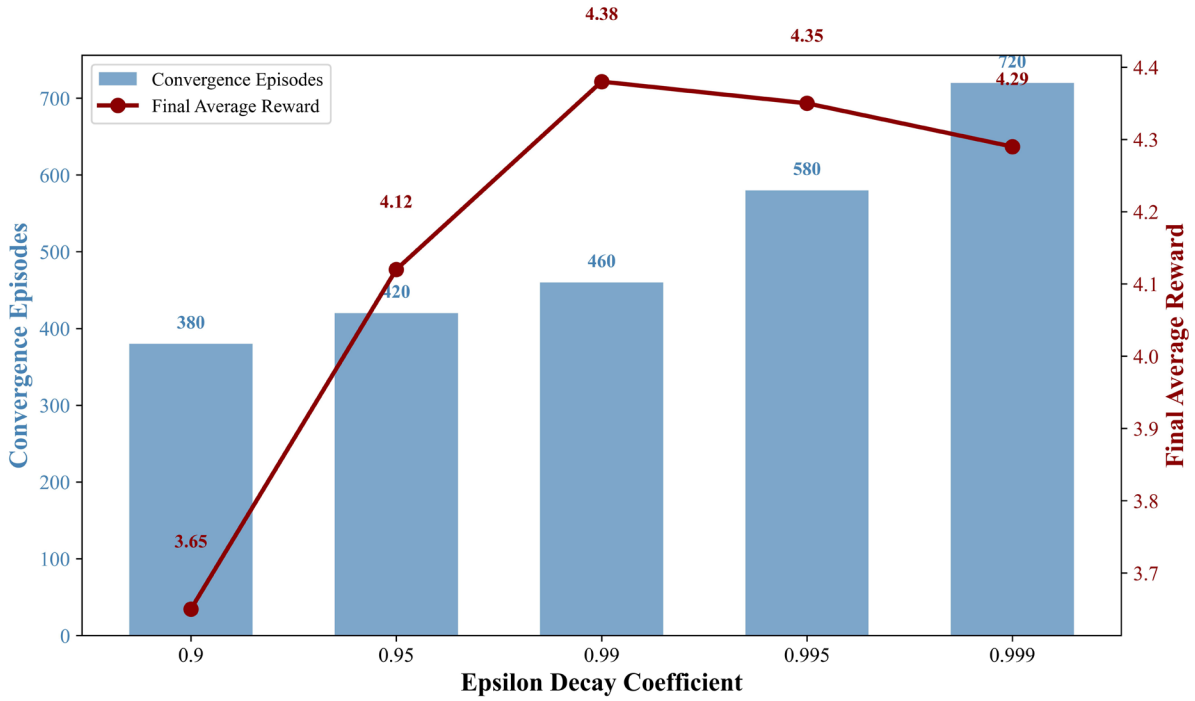

Figure 2. The influence of exploration rate attenuation coefficient on convergence speed and final reward

For the experience replay priority coefficient α, the experiment varied it from 0 (i.e., uniform sampling) to 1.5 with a step size of 0.25. The average skill acquisition time and the number of pre-requisite violation occurrences were recorded. The results are shown in Table 3. When $\alpha = 0$, it degenerates to uniform sampling. The average skill acquisition time is 38.9 steps and the number of pre-requisite violation occurrences is 1.2 times (due to the uneven sampling, the learning of the constraint samples is insufficient, and a few missed violations still occur). As α increases to 1.0, both indicators continue to improve, reaching the optimal value at $\alpha = 1.0$ (32.4 steps, 0 violations). Continuing to increase to 1.25 and 1.5, the performance shows a slight decrease (33.1 steps, 33.7 steps). This is because the excessively high priority weight causes the high-difficulty transition samples to be over-sampled, resulting in a sharp decrease in the diversity of the experience replay pool, and causing deviations in the Q-network's fitting of other conventional transfer patterns. Therefore, $\alpha = 1.0$ was selected as the final setting.

Table 3. The impact of experience replay priority coefficient α on learning performance

Priority coefficient α	Average time for skill acquisition (steps)	Number of violations of preposition usage rules	Effective diversity index of the experience replay pool
0.00	38.9	1.2	0.87
0.25	36.4	0.8	0.82
0.50	35.0	0.4	0.76
0.75	33.7	0.1	0.71
1.00	32.4	0.0	0.68
1.25	33.1	0.0	0.52
1.50	33.7	0.0	0.41

To verify the adaptability of Art-DQN to learners with different initial levels of skills, the learners in the test set were divided into three groups based on the number of skills they initially mastered (calculated from the manually annotated pre-tests): the beginner group (0-8 skills, 62 people), the advanced group (9-20 skills, 58 people), and the advanced group (21 or more skills, 55 people). Art-DQN and the strongest baseline PPO were run separately for each group, and the path redundancy and the rate of achieving the target skills were recorded. The results are shown in Figure 3.

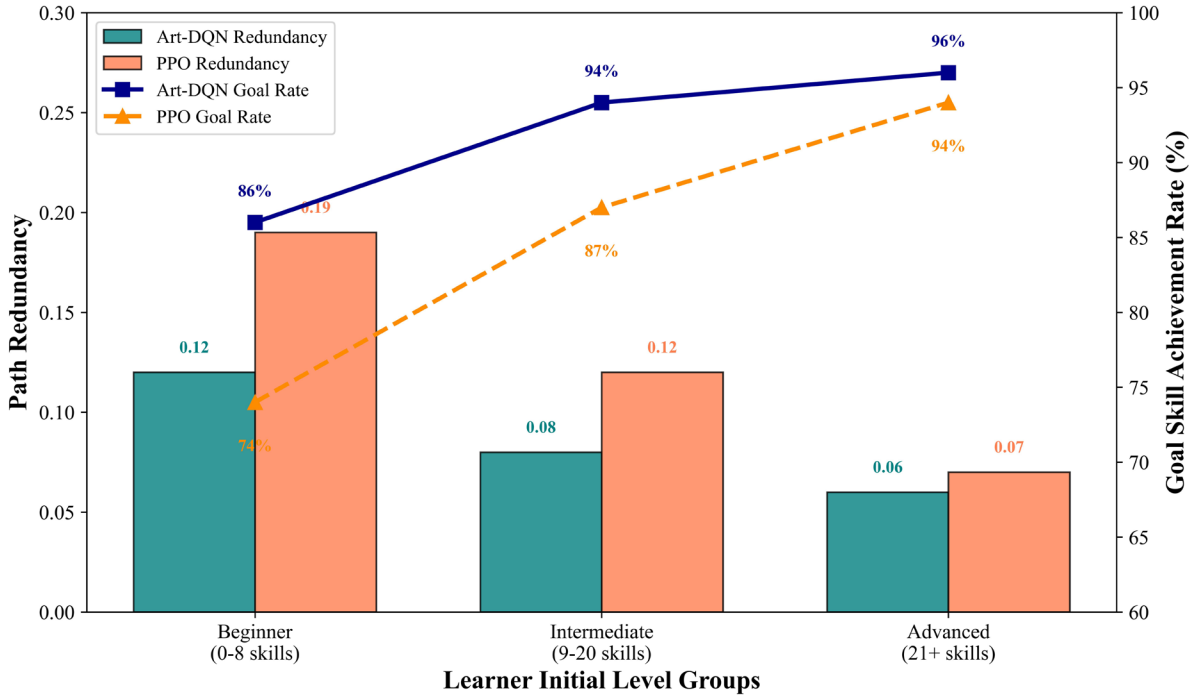

Figure 3. Comparison of path quality among different initial level learner groups

The horizontal axis represents the learner groups, while the left side of the vertical axis shows the path redundancy (bar chart), and the right side shows the target skill achievement rate (line chart). For the beginner group, the path redundancy of Art-DQN is 0.12, and that of PPO is 0.19; the target skill

achievement rate of Art-DQN is 86%, and that of PPO is 74%. For the advanced group, the gap between the two decreases but Art-DQN still has the advantage (redundancy 0.08 vs 0.12, achievement rate 94% vs 87%). For the high-level group, the performance of both is close (redundancy 0.06 vs 0.07, achievement rate 96% vs 94%). This indicates that Art-DQN is most effective for learners with weak initial knowledge reserves, as the skill topology constraint mechanism can effectively guide them to avoid skipping levels; while for advanced learners, due to the larger set of possible actions and the mastery of skills, the differences among the methods naturally decrease. Overall, Art-DQN maintains the lowest path redundancy and the highest achievement rate under all three distributions, demonstrating excellent robustness.

4. ANALYSIS OF ALGORITHM EFFICIENCY AND COMPLEXITY

The computational efficiency and resource consumption of the algorithm in actual deployment are the key indicators for evaluating its usability. This section conducts a quantitative analysis of Art-DQN from two dimensions: time complexity and space complexity, and compares it with the standard DQN and PPO algorithms. All complexity analyses are based on the following unified settings: the number of nodes in the skill topology graph $N = 48$, the embedding dimension of the graph $d' = 64$, the length of the behavior sequence $L = 10$, the hidden layer dimension of the LSTM $d_s = 64$, the size of the action space $M = 120$ (corresponding to 120 recommended learning units), the maximum capacity of the experience replay pool $|\mathcal{D}|_{\max} = 100000$, the output dimension of the dual-channel fusion layer $d_{\text{fuse}} = 128$, and the subsequent fully connected layers each contain 256 and 128 neurons, and the final Q-value output layer dimension is M.

In terms of time complexity, the main consideration is the computational cost of single-step decision-making (i.e., calculating the Q-values of all actions after a given state s_t and selecting the action). The single-step decision-making of Art-DQN can be divided into three parts: the forward calculation of the graph attention network, the forward calculation of the LSTM behavior encoding, and the forward calculation of the dual-channel fusion and the subsequent fully connected network. The graph attention network consists of two layers of GAT, and the computational complexity of each layer for each node is $\mathcal{O}(N \cdot d \cdot d' + |\mathcal{E}| \cdot d')$, where $d = 128$ is the initial node embedding dimension, and $|\mathcal{E}|$ is the number of directed edges (in this topology graph, it is approximately of the order of $3N$). Since the graph structure is static and the learner only needs to calculate the skill enhancement features g_t when making a decision, and the output of GAT can be pre-computed once and cached, the pooling calculation in the actual decision-making only requires weighted pooling based on the current vector m_t, which has a computational complexity of $\mathcal{O}(N \cdot d')$. Therefore, the single-step amortized complexity of the GAT part can be considered as $\mathcal{O}(N \cdot d')$. The LSTM behavior encoding part needs to handle a sequence of length $L = 10$, and the computational complexity of each time step is $\mathcal{O}(d_s \cdot (d_e + d_s))$, where $d_e = 32$ is the action embedding dimension. Since the sequence length is fixed and small, the complexity of this part can be considered as a constant $\mathcal{O}(L \cdot d_s^2)$. The computational cost of the dual-channel fusion and the subsequent fully connected network is mainly determined by matrix multiplication. Let the encoding dimensions of the two branches before fusion be both $d_{\text{enc}} = 64$, the parameter quantity of the fusion layer is $(d_{\text{enc}} \times d_{\text{fuse}}) \times 2 + d_{\text{fuse}} \approx 16448$, and the computational cost is approximately $\mathcal{O}(d_{\text{enc}} \cdot d_{\text{fuse}})$. The computational costs of the subsequent two fully connected layers are $\mathcal{O}(d_{\text{fuse}} \cdot 256)$ and $\mathcal{O}(256 \cdot 128)$, respectively, as well as the output layer $\mathcal{O}(128 \cdot M)$. Summarizing all these parts, the total time complexity of Art-DQN's single-step decision-making can be approximately expressed as:

$$T_{\text{Art-DQN}} = \mathcal{O}(N \cdot d' + L \cdot d_s^2 + d_{\text{enc}} \cdot d_{\text{fuse}} + d_{\text{fuse}} \cdot 256 + 256 \cdot 128 + 128 \cdot M) \quad (16)$$

Substitute typical values: $N = 48, d' = 64, L = 10, d_s = 64, d_{\text{enc}} = 64, d_{\text{fuse}} = 128, M = 120$. The calculation yields each component to be approximately 3072, 40960, 8192, 32768, 32768, 15360. The dominant terms are $L \cdot d_s^2 \approx 40960$ and $128 \cdot M \approx 15360$. The total number of floating-point operations is approximately 1.4×10^5. In contrast, the state of the standard DQN only uses the skill mastery vector $\mathrm{m}_t \in \mathbb{R}^{48}$ (or a simple multi-layer perceptron encoding), without the GAT and LSTM modules. Its single-step time complexity is:

$$T_{\text{DQN}} = \mathcal{O}(N \cdot d_{\text{fc1}} + d_{\text{fc1}} \cdot d_{\text{fc2}} + d_{\text{fc2}} \cdot M) \quad (17)$$

Here, $d_{\text{fc1}} = 128, d_{\text{fc2}} = 64$. The computational cost is approximately $48 \times 128 + 128 \times 64 + 64 \times 120 = 6144 + 8192 + 7680 = 22016$. As a strategy optimization algorithm, PPO only requires a single-step decision to calculate the action probability distribution through the policy network once, without using the target network or experience replay sampling. The policy network usually adopts a fully connected structure similar to the standard DQN, with a time complexity similar to DQN, approximately $\mathcal{O}(N \cdot d_{\text{pol}} + d_{\text{pol}}^2)$, where $d_{\text{pol}} = 64$, which is approximately $48 \times 64 + 64^2 = 3072 + 4096 = 7168$. However, due to the fact that PPO needs to collect multiple-step trajectories and perform multiple gradient updates (usually 10 epochs) during training, the time complexity of its training stage is much higher than that of a single-step decision.

In terms of space complexity, it mainly examines the storage of model parameters and the occupation of the experience replay pool. The total parameter quantity of Art-DQN includes GAT parameters, LSTM parameters, dual-channel fusion layer parameters, and fully connected network parameters. The total parameters of the GAT layer are approximately $2 \times (d \cdot d' \cdot 2 + 2d')$ (the parameters of the attention mechanism are two sets of transformations), substituting $d = 128, d' = 64$ gives approximately 32768. The LSTM parameter quantity is $4 \times (d_e \cdot d_s + d_s^2 + d_s)$, where $d_e = 32, d_s = 64$, calculated to be approximately 4 × (2048 + 4096 + 64) = 24832. The parameters of the dual-channel fusion and fully connected network are approximately 16448 + (128×256) + (256×128) + (128×120) = 16448 + 32768 + 32768 + 15360 = 97344. In addition, there is the target network (with the same structure as the main network and the same parameter quantity) and the optimizer state (Adam optimizer saves the first and second moments, the parameter quantity is twice that of the model parameters). Therefore, the total model parameter quantity of Art-DQN is approximately $(32768 + 24832 + 97344) \times 3 = 464832$ (main network + target network + optimizer state × 2 approximately). The experience replay pool stores that each sample contains the state feature g_t (64-dimensional floating point), the original action sequence before the behavior sequence encoding (10 integers), the action a_t (integer), the reward r_t (floating point), the next state feature (64-dimensional floating point), and the stop flag (Boolean). Each sample takes up $(64 + 10 + 1 + 1 + 64 + 1) \times 4$ bytes (4 bytes for floating-point numbers) ≈ 560 bytes. When the experience replay pool capacity is set to 100000, the memory usage is about 56 MB, and the model parameters are about 4.5 MB (464832 parameters, 4 bytes per float), the total space usage is about 60.5 MB.

Table 4 summarizes the comparison of time complexity and space complexity between Art-DQN and standard DQN, and PPO. It should be noted that the data listed in the table are theoretical estimates for the single-step decision-making inference stage, and the training time cost of PPO in the actual training stage is usually 2-5 times higher than that of DQN-like algorithms. From the table, it can be seen that the single-step inference complexity of Art-DQN is approximately 6.4 times that of standard DQN, but it is still within an acceptable range (modern CPUs can complete it in milliseconds). The total

number of parameters is approximately 2.6 times that of the standard DQN, mainly coming from the dual-channel feature fusion module. PPO has the highest inference efficiency, but it requires maintaining both the old policy network and the new policy network simultaneously during the training stage, and performs multiple epochs of updates. The actual training memory usage is similar to or higher than that of Art-DQN.

Table 4. Comparison of time complexity and space complexity of three algorithms (inference stage)

Algorithm	Single-step floating-point operation count (FLOPs)	Relative to the DQN multiple	Model parameter quantity (ten thousand)	Experience replay pool usage (MB, capacity 100,000)	Total memory usage (MB)
Standard DQN	Approximately 2.20×10^4	1.0×	Approximately 1.78	56	Approximately 57.1
PPO	Approximately 7.17×10^3	0.33×	Approximately 1.20	Not relying on (track cache of approximately 5MB)	Approximately 9.8
Art-DQN	Approximately 1.40×10^5	6.36×	Approximately 4.65	56	Approximately 60.5

Although Art-DQN has a higher single-step inference complexity than standard DQN and PPO, considering the requirements for real-time performance in the scenario of art learning path recommendation are not overly strict (the response time for each recommendation is within 500 milliseconds and meets the user experience), and the learning efficiency improvement brought by Art-DQN (as shown in Section 3.3, the skill acquisition time is reduced by 22.3%) can significantly reduce the overall number of computational interactions for learners, from an end-to-end perspective, the overall computational cost may actually be lower. Moreover, during actual deployment, optimization measures such as model quantization, knowledge distillation, or pre-computing and offline storing the output of the graph attention part in advance can be adopted to further reduce the computational cost of online inference. In summary, Art-DQN exchanges a moderate computational and storage cost for significant improvement in recommendation quality, and its complexity is feasible and acceptable in practical applications.

5. DISCUSSION

The Art-DQN algorithm proposed in this paper demonstrates significant advantages over general recommendation methods in the task of recommending art learning paths. Its core lies in deeply embedding the progressive rules of art-specific skills into the decision-making process. Traditional collaborative filtering or content-based recommendation methods often only focus on learners' historical preferences or skill mastery levels, ignoring the inherent dependency structure among art skills - for instance, before learning a portrait painting, one must master the skills of facial proportions, facial structure, and light and shadow processing, etc. Art-DQN constructs a directed skill topology graph and enforces pre-requisite dependency constraints in the exploration strategy, fundamentally avoiding the inefficacy of recommending jumping content that leads to invalid learning paths. Experimental results show that the number of pre-requisite violations for Art-DQN is zero, while standard DQN averages 3.2 violations. This directly proves the effectiveness of the topological

constraints. In contrast, PPO and standard DQN can indirectly penalize violations through the reward function, but they cannot completely eliminate them because purely data-driven methods are difficult to learn sparse and absolute dependency relationships. Moreover, the redundancy of the path for Art-DQN is 0.09, which is much lower than 0.21 of the knowledge graph method, indicating that the reinforcement learning framework not only ensures the legality of the path but also optimizes the active selection of the most compact sequence of skill advancement rather than mechanically traversing all nodes in the topological order. This advantage is particularly evident in novice learners, with a target skill achievement rate 12 percentage points higher than PPO, because learners with weak initial knowledge are most likely to experience frustration and even give up learning due to skipping recommendations, while the constraint mechanism of Art-DQN provides them with stable scaffolding-like guidance.

Any method has its applicable scope and limitations. One limitation of Art-DQN concerns path stability for users with extremely low learning engagement. Low-engagement users refer to those who have less than two learning interactions per week and each learning session lasts no more than 15 minutes. In the data set of this paper, such users account for approximately 12% of the total sample. Analysis reveals that when learners have long learning intervals, the skill forgetting model (see Section 3.1) significantly reduces the effective level of the skills they have mastered, but the skill mastery vector m_t in Art-DQN's state representation is updated based on self-assessment scales or test results, while low-engagement users often do not re-assess their skills before each regression learning, resulting in a deviation in the state representation - the system mistakenly believes that the learners still possess certain skills, while in fact these skills have partially deteriorated due to forgetting. This state estimation error causes the algorithm to recommend subsequent skills beyond the learner's true ability range, leading to unstable paths, manifested as a decrease in completion rate and frequent re-learning behaviors. Another factor contributing to instability is the sparse behavioral sequence of low-engagement users, and the LSTM module has difficulty extracting effective temporal patterns from scarce historical actions, and the dual-channel fusion mechanism actually degenerates into a single-channel (only relying on skill features), weakening the algorithm's personalized ability. This limitation suggests that for low-activity scenarios, relying solely on interaction log-driven reinforcement learning methods may not be the best choice, and more lightweight cold-start strategies or active intervention mechanisms need to be introduced.

To address these limitations and for broader application expansion, future research can seek breakthroughs in the direction of path migration across art styles. The current skill topology graph constructed by Art-DQN is designed for a specific art category (with Western painting as the main reference). When learners have a solid foundation in oil painting and then switch to learning Chinese painting, there are significant differences in the skill dependency structures of the two fields - Chinese painting emphasizes ink and brushwork flavor, blank space conception, and calligraphy brushstrokes, while oil painting's techniques such as color mixing and brush texture are not completely interchangeable at the underlying level. However, certain high-level cognitive abilities such as observation methods, composition thinking, and aesthetic judgment have the potential for cross-style transfer. Introducing a meta-learning mechanism, especially model-agnostic meta-learning (MAML) or meta-reinforcement learning framework, can enable the algorithm to quickly adapt to the skill topology structure of a new field with a small amount of Chinese painting learning samples. Specifically, the meta-training stage can be pre-executed on multiple art styles (oil painting, watercolor, sketch, Chinese painting) learning datasets, with each style corresponding to a skill topology graph and an Art-DQN model. The goal of meta-learning is to learn a shared initialization parameter, so that when a new style

appears, only a small number of learner interaction trajectories are needed to quickly converge to an effective path recommendation strategy. This cross-style transfer not only solves the problem of sparse data in a single domain but also provides a unified recommendation framework for interdisciplinary art education (such as the comprehensive artistic literacy cultivation of design majors). Additionally, meta-learning can be used to solve the problem of low participation users: using the interaction trajectories of high-activity users as the source task, through meta-learning to extract general learning path patterns, and then adapting them for low-participation users with few samples, thereby improving their path stability. Another direction worth exploring is combining the large language models from the natural language processing field with Art-DQN, using the large language model to automatically parse open-ended art learning goals (such as "I want to learn the light and shadow expression techniques of Impressionism") and break them down into sub-goal sequences on the skill topology graph, and then the reinforcement learning module executes fine-grained path optimization, thereby further enhancing the algorithm's ability to respond to individual learning needs.

6. CONCLUSION

This paper addresses the challenges in online art education, such as highly nonlinear learning paths, complex skill dependency relationships, and significant individual differences. It proposes an individualized art learning path recommendation algorithm called Art-DQN based on the improved deep Q-network. The core innovations of this algorithm are reflected in three aspects: Firstly, an art skill directed topological graph is constructed, and a graph attention network is introduced to enhance the structured representation of learners' skill mastery status, enabling the state space to capture high-order correlation information among skills; Secondly, a dual-channel feature fusion Q-network architecture is designed, integrating skill graph features and learning behavior sequence features in the fusion layer, significantly improving the modeling ability for learners' long-term learning patterns; Finally, an ε-greedy exploration strategy based on skill topological constraints and a priority experience replay mechanism are proposed, fundamentally eliminating invalid recommendations that violate the pre-existing dependency relationships, and accelerating the learning convergence of key patterns through the prioritized sampling of high-difficulty skill transition samples.

Through thorough experimental verification in an art learning environment containing real course logs and simulation extended data, Art-DQN significantly outperforms random recommendations, knowledge graph-based methods, standard DQN, and PPO algorithms in multiple core indicators. Specifically, the average skill mastery time of Art-DQN is reduced by 22.3% compared to standard DQN, the target skill achievement rate is increased to 92%, and throughout the test, there are zero violations of the prerequisite dependencies. Ablation experiments further confirm the independent contributions of the dual-channel feature fusion module and dynamic reward weight to performance. Removing the fusion module leads to a 13.6% increase in skill mastery time. Parameter sensitivity analysis determines that the exploration rate decay coefficient of 0.99 and the priority sampling coefficient of 1.0 are the optimal settings. Robustness tests show that the algorithm has the most significant advantage for beginner-level learners, with a target skill achievement rate 12 percentage points higher than PPO. Complexity analysis indicates that Art-DQN achieves significant recommendation quality improvement with moderate computational and storage costs, and the single-step inference floating-point operation volume is approximately 6.4 times that of standard DQN, while fully meeting the real-time requirements for practical deployment.

From an application value perspective, Art-DQN provides a technical solution for intelligent

recommendation systems in the art education field, which takes into account both structural constraints and personalized adaptation. Compared with traditional recommendation methods, this algorithm not only recommends appropriate next learning content based on the current level of learners, but also actively plans a complete set of efficient and compact skill advancement paths through long-term rewards, especially suitable for art disciplines with clear skill hierarchical structures (such as painting, music, design, etc.). For educational platforms, this algorithm is expected to reduce the number of ineffective learning attempts and frustration of learners, increase course completion rates and user retention; for individual learners, it can achieve high-level art goals in a shorter time while ensuring the completeness of the knowledge system. Future work will focus on achieving cross-art style (such as from oil painting to traditional Chinese painting) path transfer capabilities through meta-learning, and exploring the combination of large language models and reinforcement learning to support more open and flexible learning goal interpretation and recommendation.

Abbreviations

DQN, Deep Q-Network;

Art-DQN, Artistic Deep Q-Network;

GAT, Graph Attention Network;

LSTM, Long Short-Term Memory;

PPO, Proximal Policy Optimization;

KG-Rec, Knowledge Graph-based Recommendation;

MDP, Markov Decision Process;

TD, Temporal Difference;

ReLU, Rectified Linear Unit;

ELU, Exponential Linear Unit;

MLP, Multilayer Perceptron;

MAML, Model-Agnostic Meta-Learning;

CPU, Central Processing Unit;

GPU, Graphics Processing Unit;

FLOPs, Floating Point Operations.

Supplementary Material

Not applicable.

Appendix

Not applicable.

Ethics approval and consent to participate.

This study did not involve human participants, animal subjects, or any data requiring ethical approval. Therefore, ethics approval and consent to participate are not applicable.

Acknowledgements

The authors would like to thank the editors of this journal and all the anonymous reviewers who provided valuable comments on this work.

Competing interests

The authors declare that they have no financial or personal relationships that may have inappropriately influenced them in writing this article.

Author contributions

All authors have read and agreed to the published version of the manuscript. The author's contributions are specified as follows: **S.S.:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization, Supervision. **Y.L.:** Resources, Data Curation, Validation, Investigation, Writing – review & editing, Project administration.

Funding information

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data availability

The data that support the findings of this study are available upon request from the corresponding authors, **Y.L.**

Disclaimer

The views and opinions expressed in this article are those of the authors and are the product of professional research. It does not necessarily reflect the official policy or position of any affiliated institution, funder, agency, or that of the publisher. The authors are responsible for this article's results, findings, and content.

Declaration of AI and AI-assisted Technologies in the Writing Process

During the writing of this article, the author used DeepSeek for spelling and grammar checking. After using this tool, the author reviewed and edited the content as needed and assumes full responsibility for the final published content.

REFERENCES

[1] Savage, T. (2024). *Teaching to the line: how do visual arts technicians in higher education conceive of their pedagogies?* (Doctoral dissertation, University for the Creative Arts). https://doi.org/10.13140/RG.2.2.18860.99203

[2] Shao, J. (2024). *Personalized Learning for Art Major Students Based on Learner Characteristics* (Doctoral dissertation, Chapman University). https://doi.org/10.36837/chapman.000560

[3] Ткач, М., Олексюк, О. М., Бобик, Л., Мимрик, М., & Вей, Л. (2024). Non-linear thinking strategies in post-non-classical higher art education: A synergistic concept. *Scientific Herald of Uzhhorod University*, *55*, 1994-2005. https://doi.org/10.54919/physics/55.2024.199kv4

[4] Li, Y., & Shi, J. (2025). Multimodal deep learning for art behavior analysis and personalized teaching path generation. *Discover Artificial Intelligence*, *5*(1), 215. https://doi.org/10.1007/s44163-025-00480-w

[5] Shokrzadeh, Z., Feizi-Derakhshi, M. R., Balafar, M. A., & Mohasefi, J. B. (2024). Knowledge graph-based recommendation system enhanced by neural collaborative filtering and knowledge graph embedding. *Ain Shams Engineering Journal*, *15*(1), 102263. https://doi.org/10.1016/j.asej.2023.102263

[6] Troussas, C., & Krouska, A. (2022). Path-based recommender system for learning activities using knowledge graphs. *Information*, *14*(1), 9. https://doi.org/10.3390/info14010009

[7] Qian, D., & Luo, W. (2026). Revolutionizing arts education through 3D virtual reality: a mixed-method analysis of its impact on sculpting and carving skills among undergraduate art students. *Educational technology research and development*, 1-36. 1001-1036 https://doi.org/10.1007/s11423-025-10584-w

[8] Du, W., Zhao, Y., Wang, Y., Wang, H., & Yang, M. (2022). Novel machine learning approach for shape-finding design of tree-like structures. *Computers & Structures*, *261*, 106731. https://doi.org/10.1016/j.compstruc.2021.106731

[9] Zhao, L. T., Wang, D. S., Liang, F. Y., & Chen, J. (2023). A recommendation system for effective learning strategies: An integrated approach using context-dependent DEA. *Expert Systems with Applications*, *211*, 118535. https://doi.org/10.1016/j.eswa.2022.118535

[10] Garg, S., & Roy, D. (2022). A birds eye view on knowledge graph embeddings, software libraries, applications and challenges. *arXiv preprint arXiv:2205.09088*. https://doi.org/10.48550/arXiv.2205.09088

[11] Wani, A. A. (2025). Comprehensive review of dimensionality reduction algorithms: challenges, limitations, and innovative solutions. *PeerJ Computer Science*, *11*, e3025.

https://doi.org/10.7717/peerj-cs.3025

[12] Wang, Z., Li, S., Liu, Q., Pan, Z., & Sun, X. (2026). MKAN-Refine: Fine-Grained Crisis Information Mining via Reliability-Aware Nonlinear Refinement and Kolmogorov–Arnold Networks. *IEEE Access*. https://doi.org/10.1109/access.2026.3688494

[13] Li, P., & Ding, Z. (2025). Application of deep learning-based personalized learning path prediction and resource recommendation for inheriting scientist spirit in graduate education. *Computer Science and Information Systems*, (00), 43-43. https://doi.org/10.2298/csis241125043l

[14] Angelaki, S., Triantafyllidis, G. A., & Besenecker, U. (2022). Lighting in kindergartens: Towards innovative design concepts for lighting design in kindergartens based on children's perception of space. *Sustainability*, *14*(4), 2302. https://doi.org/10.3390/su14042302

[15] Newbury, R., Gu, M., Chumbley, L., Mousavian, A., Eppner, C., Leitner, J., ... & Cosgun, A. (2023). Deep learning approaches to grasp synthesis: A review. *IEEE Transactions on Robotics*, *39*(5), 3994-4015. https://doi.org/10.48550/arXiv.2207.02556

[16] Bai, Y., Liu, Z., Guo, T., Hou, M., & Xiao, K. (2025). Prerequisite relation learning: a survey and outlook. *ACM Computing Surveys*, *57*(11), 1-28. https://doi.org/10.1145/3733593

[17] Wang, Z., Yan, W., Zeng, C., Tian, Y., & Dong, S. (2023). A unified interpretable intelligent learning diagnosis framework for learning performance prediction in intelligent tutoring systems. *International Journal of Intelligent Systems*, *2023*(1), 4468025. https://doi.org/10.1155/2023/4468025

[18] Winget, M., & Persky, A. M. (2022). A practical review of mastery learning. *American journal of pharmaceutical education*, *86*(10), ajpe8906. https://doi.org/10.5688/ajpe8906

[19] Yu, S., Zeng, Y., Yang, F., & Pan, Y. (2024, March). Causal-driven skill prerequisite structure discovery. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 38, No. 18, pp. 20604-20612). https://doi.org/10.1609/aaai.v38i18.30046

[20] Vrahatis, A. G., Lazaros, K., & Kotsiantis, S. (2024). Graph attention networks: a comprehensive review of methods and applications. *Future Internet*, *16*(9), 318. https://doi.org/10.3390/fi16090318

[21] Yu, X., Yang, S., Wang, Z., Song, S., Ma, H., Cao, Z., & Zhang, X. (2025, July). LIGHT: Enhancing Learning Path Recommendation via Knowledge Topology-Aware Sequence Optimization. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 306-315). https://doi.org/10.1145/3726302.3730022

[22] Amin, S., Uddin, M. I., Alarood, A. A., Mashwani, W. K., Alzahrani, A., & Alzahrani, A. O. (2023). Smart E-learning framework for personalized adaptive learning and sequential path recommendations using reinforcement learning. *IEEe Access*, *11*, 89769-89790. https://doi.org/10.1109/access.2023.3305584

[23] Malashin, I., Tynchenko, V., Gantimurov, A., Nelyub, V., & Borodulin, A. (2024). Applications of long short-term memory (LSTM) networks in polymeric sciences: A review. *Polymers*, *16*(18), 2607. https://doi.org/10.3390/polym16182607

[24] Xu, W., He, J., Li, W., He, Y., Wan, H., Qin, W., & Chen, Z. (2023). Long-short-term-memory-based deep stacked sequence-to-sequence autoencoder for health prediction of industrial workers in

closed environments based on wearable devices. *Sensors*, *23*(18), 7874. https://doi.org/10.3390/s23187874

[25] Jaramillo-Martínez, R., Chavero-Navarrete, E., & Ibarra-Pérez, T. (2024). Reinforcement-learning-based path planning: A reward function strategy. *Applied Sciences*, *14*(17), 7654. https://doi.org/10.3390/app14177654

[26] Gallici, M., Fellows, M., Ellis, B., Pou, B., Masmitja, I., Foerster, J., & Martin, M. (2025, May). Simplifying deep temporal difference learning. In *International Conference on Learning Representations* (Vol. 2025, pp. 78148-78190). https://doi.org/10.48550/arXiv.2407.04811

[27] Hong, Z. W., Kumar, A., Karnik, S., Bhandwaldar, A., Srivastava, A., Pajarinen, J., ... & Agrawal, P. (2023). Beyond uniform sampling: Offline reinforcement learning with imbalanced datasets. *Advances in Neural Information Processing Systems*, *36*, 4985-5009. https://doi.org/10.48550/arXiv.2310.04413

[28] Lockwood, O., & Si, M. (2022, October). A review of uncertainty for deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment* (Vol. 18, No. 1, pp. 155-162). https://doi.org/10.1609/aiide.v18i1.21959

[29] Petravičius, T. (2023). *Research and analysis of reinforcement learning methods in OpenAI Gym environment* (Doctoral dissertation, Kauno technologijos universitetas.).

[30] Yadav, R. K. (2025). Modeling Memory Retention with Ebbinghaus's Forgetting Curve and Interpretable Machine Learning on Behavioral Factors. *Authorea Preprints*. https://doi.org/10.36227/techrxiv.174495325.58680708/v1

[31] De Melo, C. M., Torralba, A., Guibas, L., DiCarlo, J., Chellappa, R., & Hodgins, J. (2022). Next-generation deep learning based on simulators and synthetic data. *Trends in cognitive sciences*, *26*(2), 174-187. https://doi.org/10.1016/j.tics.2021.11.008

[32] Ghorbani, Z., Mirebeigi-Jamasbi, S. S., Hassannia Dargah, M., Nahvi, M., Hosseinikhah Manshadi, S. A., & Akbarzadeh Fathabadi, Z. (2025). A novel deep learning-based model for automated tooth detection and numbering in mixed and permanent dentition in occlusal photographs. *BMC Oral Health*, *25*(1), 455. https://doi.org/10.1186/s12903-025-05803-y

[33] Zhang, Z. (2026). Dynamic Pricing Strategy Optimization Based on a Reinforcement Learning PPO Algorithm: An Empirical Study on Ride-Hailing Platforms. *Journal of Organizational and End User Computing (JOEUC)*, *38*(1), 1-43. https://doi.org/10.4018/JOEUC.406688

APA Citation

Sun, S., & Li, Y. (2026). Research on Personalized Art Learning Path Recommendation Algorithm Based on Reinforcement Learning. *Journal of Discovery Core, 1*(1), 55-76. **https://doi.org/10.67541/jdc2603**