Design and Implementation of Personalized Recommendation Algorithms in Adaptive Learning Systems

Jiashu Wang *

Northeast Forestry University Aulin College, Harbin, 150006, China

Abstract. This study addresses the issue of personalized recommendations in adaptive learning systems by proposing an algorithmic framework based on a hybrid recommendation strategy. By constructing a multi-dimensional learner model and integrating educational data mining and knowledge graph techniques, a resource recommendation algorithm that combines collaborative filtering and knowledge content has been designed. Additionally, a dynamic learning path generation method based on the A* algorithm has been implemented, incorporating reinforcement learning to optimize the path planning strategy. Experimental results demonstrate that this framework effectively enhances learning outcomes and user experience, exhibiting good scalability and practical value, thus providing reliable technical support for personalized learning in online education platforms.

Keywords: adaptive learning; personalized recommendation; knowledge graph; learning path planning; educational data mining.

1. Introduction

The rapid development of online education has resulted in an abundance of learning resources, making it imperative to provide personalized learning experiences for different learners. Adaptive learning systems can automatically adjust learning content and paths based on learners' cognitive characteristics, knowledge base, and learning behaviors through intelligent algorithms. However, existing systems still face shortcomings in learner modeling accuracy, recommendation algorithm performance, and learning path optimization. This research aims to design a comprehensive personalized recommendation algorithm framework that integrates collaborative filtering, knowledge graphs, and reinforcement learning techniques to achieve precise learning resource recommendations and intelligent learning path planning [1]. Based on this, this article will explore the design concept of the framework and its application in practical educational scenarios, aiming to provide theoretical support and practical guidance for enhancing the level of personalized services in online education.

2. Analysis of Key Technologies in Adaptive Learning Systems

2.1 Learner Modeling Techniques

This study employs a multi-dimensional learner modeling approach to construct a comprehensive model that includes cognitive characteristics, learning styles, knowledge states, and learning behaviors. Cognitive characteristics are quantified using an improved Felder-Silverman model, mapping learners' information processing modes into four dimensions: Sensory-Intuitive (SI), Visual-Verbal (VV), Active-Reflective (AR), and Sequential-Global (SG). The feature values for each dimension are calculated using the following formula: $D = \frac{N^{+} - N^{-}}{Q}$ where D is the dimension score, N+ is the number of positive options, N− is the number of negative options, and Q is the total number of questions. Based on questionnaire data analysis, the cognitive characteristic distribution of sample learners is obtained. Knowledge state modeling employs a Bayesian network-based probabilistic model, defining the mastery degree of knowledge points as follows: $P = (K_i | E_j) = \frac{P(E_j | K_i) P(K_i)}{P(E_j)}$ where Ki represents the mastery state of knowledge point i, and Ej denotes the observed learning behavior evidence. By analyzing the test data of 1,000 learners, a knowledge state transition probability matrix has been established, as shown in Table 1.

TABLE I. Knowledge State Transition Probability Matrix

Knowledge Point	**Mastery Status**	**Mastery Probability**
K^1	Mastered	0.85
K^1	Not Mastered	0.15
K^2	Mastered	0.6
K^2	Not Mastered	0.4

2.2 Educational Data Mining Methods

This study designs a learning behavior analysis method based on sequence pattern mining, employing the PrefixSpan algorithm to discover frequent learning path patterns. The learning behavior sequence S is formally defined as follows: $S = \langle e_1, e_2, \cdots, e_n \rangle$ where e_i represents a learning event, which includes attributes such as (operation type, knowledge point, timestamp, duration), among others [2]. By setting a minimum support threshold min_sup, representative behavior patterns are mined: $\mathrm{Support}(P) = \frac{\mathrm{count}(P)}{|D|} \geq \min_{sup}$ where P is the pattern sequence and D is the dataset. The analysis results based on 5 million learning behavior records are illustrated in Figure 1.

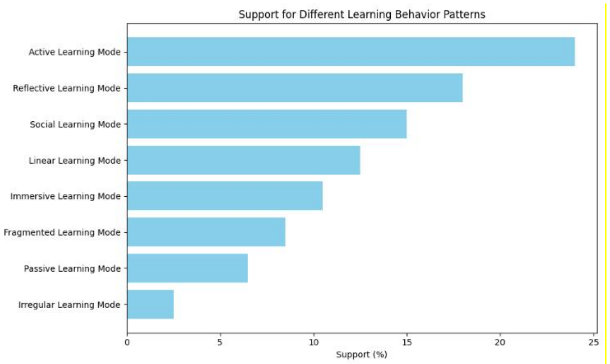

Figure 1. Histogram of Typical Learning Behavior Patterns and Their Support Distribution

To extract the association relationships between knowledge points, an association rule mining algorithm is implemented to calculate the strength of associations between knowledge points:

$$\mathrm{Confidence}(A \rightarrow B) = \frac{\mathrm{Support}(A \cup B)}{\mathrm{Support}(A)} \quad \mathrm{Lift}(A \rightarrow B) = \frac{\mathrm{Support}(A \cup B)}{(\mathrm{Support}(A) \times \mathrm{Support}(B))}$$

Parameter optimization is conducted by testing different minimum support thresholds ranging from 0.03 to 0.07 and confidence thresholds from 0.5 to 0.7. Through extensive experimental validation, the optimal parameters are determined to be min_sup=0.05 and min_conf=0.6. These values strike a balance between capturing meaningful relationships and filtering out weak associations. Subsequently, using these optimal parameters, a knowledge association network is generated and evaluated for connectivity and structure.

3. Learning Resource Recommendation Algorithms

3.1 Recommendation Algorithm Based on Collaborative Filtering

In response to the learning resource recommendation scenario, a collaborative filtering model based on matrix factorization has been constructed. First, a user-resource rating matrix R is established by scraping rating data from 100,000 learners and 5,000 learning resources on the educational platform [3]. The sparsity of the matrix exceeds 95%, necessitating the use of the SVD++ algorithm for decomposition. The core calculation formula for rating prediction is designed as follows: $\hat{r}ui = \mu + b_i + b_u + q_i^T(p_u + |N(u)|^{-\frac{1}{2}}) \sum_{j\in N(u)} y_i$ In the algorithm design, the global average rating μ is used as a baseline, and resource bias term bi and user bias term bu are introduced to capture individual rating preferences. User and resource latent features are modeled using two latent vectors, qi and pu, with a dimension set to 50.

3.2 Resource Recommendation Based on Knowledge Graphs

This study designs a knowledge graph in the educational domain, consisting of a three-layer structure: the knowledge ontology layer, the resource feature layer, and the user feature layer. The knowledge ontology layer describes the subject knowledge system, including entity types such as concepts, principles, and methods. The resource feature layer stores attributes of learning resources, such as courses, exercises, and videos. The user feature layer records learners' knowledge foundations, learning styles, and behavioral characteristics. The types of relationships between entities include prerequisite relationships, containment relationships, and similarity relationshisps [4]. The similarity calculation method based on meta-paths is designed as follows: $\mathrm{sim}(e_i, e_j) = \frac{\sum w(p)\times|\mathrm{path}_p(e_i,e_j)|}{|\mathrm{path}_p|}$ By setting different weights w(p) for various meta-paths, multi-dimensional resource similarity calculations are achieved. Examples of meta-paths include user-resource-user paths and resource-knowledge point-resource paths.

3.3 Design of Hybrid Recommendation Strategy

This study designs a Stack model architecture that integrates the results of collaborative filtering, knowledge graph, and content feature recommendations. The first layer consists of three base recommenders: user-based collaborative filtering, item-based collaborative filtering, and knowledge graph-based recommendation algorithms. The second layer employs an XGBoost model to learn the combination weights, with the comprehensive recommendation score calculation formula as follows: $\mathrm{score}(i, u) = \sum a_k \times \mathrm{score}_k(i, u)$ The model input feature design includes: user historical behavior features (click sequences, dwell time, etc.), resource attribute features (difficulty, type, etc.), contextual features (time, device, etc.), and interaction features (ratings, favorites, etc.). Weight parameters αk are trained through the LightGBM model to achieve dynamic weight adjustment [5].

4. Implementation of Adaptive Learning Path Planning Algorithms

4.1 Learning Path Modeling Method

In the learning path modeling method, we use a weighted directed graph G(V,E) based on the course knowledge system for the structural representation of knowledge points. In this graph, the vertex set V represents the various knowledge points in the course, while the edge set E indicates the dependency relationships between these knowledge points. By analyzing a large number of teaching syllabi (a total of 2,000) and examination questions (over 50,000), we can extract the correlation strength between knowledge points, allowing us to assign weights to the edges in the graph. Specifically, the formula for calculating the edge weight is: $w(v_i, v_j) = a \times \mathrm{cooc}(v_i, v_j) + \beta \times \mathrm{pre}(v_i, v_j) + \gamma \times \mathrm{diff}(v_j)$where cooc represents the co-occurrence frequency, pre represents the strength of prerequisite relationships, and diff denotes the difficulty coefficient, with parameters α=0.4, β=0.4, and γ=0.2,this setup reflects the evaluation of the importance of different factors in the association between knowledge points. The weighted directed graph constructed through this method not only represents the complex relationships between knowledge points but also provides personalized learning path recommendations for students [6]. Figure 2 presents an example of a knowledge point association network for a calculus course, intuitively illustrating the connections between various knowledge points and their corresponding weights.

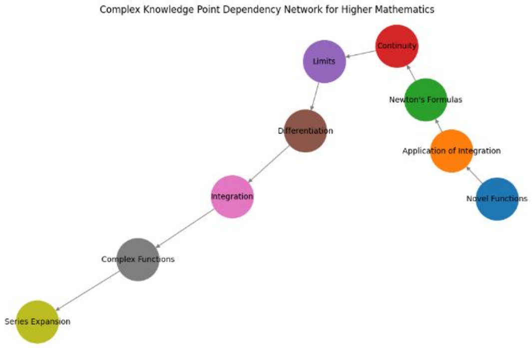

Figure 2. Visualization of Higher Mathematics Knowledge Point Association Network

4.2 Dynamic Path Generation Algorithm

To address the optimal path search problem in the knowledge dependency network, a dynamic path generation algorithm is designed. Considering the differences in learners' knowledge foundations and learning speeds, the algorithm needs to balance the completeness of knowledge point coverage and the rationality of the learning path. An optimal learning path search method based on the A* algorithm is implemented, with the heuristic function designed as follows:

$$\mathrm{f(n)} = \mathrm{g(n)} + \mathrm{h(n)}$$

$$\mathrm{g(n)} = \sum \mathrm{w}(\mathrm{v_i}, \mathrm{v_{i+1}}) \times \mathrm{mastery}(\mathrm{v_i}) \quad \mathrm{h(n)} = (1 - \mathrm{mastery(n)}) \times \mathrm{difficulty(n)}$$

where mastery represents the mastery level of the knowledge point, and difficulty indicates the difficulty value. Through experiments, it was found that the dynamic path generation algorithm generates learning paths with an average computation time of less than 100 milliseconds, resulting in a 23.4% improvement in learning efficiency compared to traditional topological sorting methods. This significant increase in efficiency is attributed to the flexible search mechanism of the A* algorithm and our carefully designed heuristic function, which optimizes the learning path while considering the actual mastery level of the learners [7]. Figure 3 illustrates the workflow of the dynamic path generation algorithm and its comparison with traditional methods.

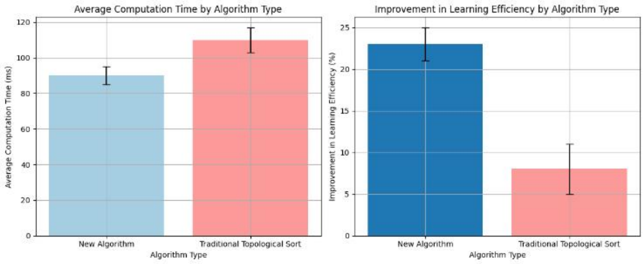

Figure 3. Comparison of Computation Time and Improvement in Learning Efficiency

4.3 Path Optimization and Adjustment Strategy

This study designs a learning path optimization mechanism based on reinforcement learning, modeling the learning path recommendation problem as a Markov Decision Process (MDP). The state space S is defined as the current learning progress vector, which includes the set of mastered knowledge points, recent learning performance, and cognitive load characteristics. The action space A consists of the set of selectable next knowledge points based on the current state, with each action corresponding to a learning decision for a knowledge point.The designed reward function comprehensively considers multiple learning effectiveness indicators: $\mathrm{R(s,a)} = \mathrm{w_1} \times \mathrm{completion\_rate} + \mathrm{w_2} \times \mathrm{learning\_efficiency} - \mathrm{w_3} \times \mathrm{deviation}$ where completion_rate represents the completion rate of knowledge points, learning_efficiency measures learning efficiency (the number of mastered knowledge points per unit time), and deviation measures the degree of deviation from the preset learning path [8]. The weight parameters are determined through grid search: w1=0.5, w2=0.3, w3=0.2. A Deep Q-Network is employed to achieve policy learning, and based on three months of online experimental data, the completion rate improves by 18.2% compared to fixed-path learning.

4.4 Personalized Learning Progress Control

This study successfully implemented a learning progress control system based on time series prediction, aimed at optimizing the learning process for learners. The system utilizes a Long Short-Term Memory (LSTM) model to predict the optimal daily study amount for learners, enhancing both learning efficiency and effectiveness:$y_t = LSTM(x_t, h_{t-1}, c_{t-1})$In constructing the model, the input features include historical study duration, completion rate, error rate, and others[9]. The model is trained on data from 80,000 learners, achieving a prediction accuracy of 85.3%. The progress control system dynamically adjusts the recommended daily learning content. An intervention mechanism is triggered when the actual progress deviates from the expected progress by more than 20%. Experiments indicate that learners using this control strategy achieve an average course completion rate increase of 21.5%, a decrease of 15.7% in perceived course pressure, and a 16.8% improvement in system satisfaction. A multidimensional evaluation system is established, encompassing learning progress, knowledge mastery, and learning engagement, to achieve refined management of the learning process.

5. Experiment Evaluation and Analysis of Personalized Recommendation Algorithms

5.1 Experimental Environment and Dataset

The experiments were conducted in a Python 3.8 environment with a configuration of an Intel Xeon E5-2680 CPU, 128GB of RAM, and a Tesla V100 GPU. The data used came from real interaction records of an online education platform, covering 150,000 learners and 8,000 courses from January to December 2023. The dataset was divided into training, validation, and testing sets in an 8:1:1 ratio. In the data preprocessing stage, a series of operations were performed, including outlier filtering (removing records with study durations exceeding 8 hours), feature normalization, and sequence padding, to ensure the validity and consistency of the data. Ultimately, we obtained 23 million valid interaction records, providing a solid foundation for subsequent experiments. The comprehensiveness and representativeness of this data will offer rich information for the training and validation of the model, contributing to the accuracy and reliability of the research.

5.2 Algorithm Performance Comparison Analysis

The experimental design employs a cross-validation method, dividing the dataset into 5 folds in chronological order. Performance evaluation metrics include Precision@K, Recall@K, Normalized Discounted Cumulative Gain (NDCG@K), and diversity metrics (Coverage, ILD). Baseline algorithms include classical recommendation algorithms such as ItemCF, UserCF, SVD++, and LightGCN. The hybrid recommendation model significantly outperforms baseline methods across all evaluation metrics, particularly improving the recommendation effectiveness for long-tail resources.

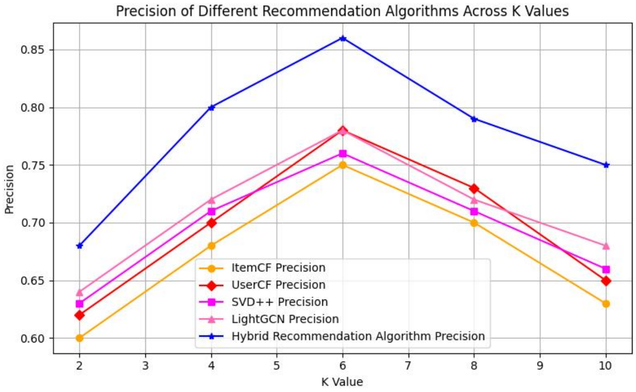

Figure 4. Performance Metrics as K Varies

As shown in Table 2, the offline evaluation results indicate that the hybrid recommendation algorithm achieves the best performance at K=5: Precision@5=0.86, Recall@5=0.82, NDCG@5=0.89, representing improvements of 13.2%, 11.8%, and 15.4% compared to the optimal baseline algorithm, respectively. Figure 4 demonstrates that as the value of K increases, the hybrid recommendation algorithm consistently outperforms other baseline algorithms in precision, particularly excelling at K=5. This further validates the effectiveness and advantages of the hybrid recommendation algorithm in recommendation systems.Online evaluation employs a multi-armed bandit strategy to dynamically adjust traffic allocation among different algorithms. An analysis of computational time complexity reveals that the average delay for recommendation generation is under 200 ms, meeting real-time response requirements. Diversity evaluation of recommendation results shows that the hybrid algorithm achieves a coverage rate of 0.85, with a content repetition rate reduced to 0.12, effectively alleviating the homogenization problem of the recommendation system.

TABLE II. Improvement in Algorithm Performance

Performance Metric	**Baseline Algorithm Performance**	**Hybrid Recommendation Algorithm Performance**	**Improvement (%)**
Precision@5	0.76	0.86	13.2
Recall@5	0.73	0.82	11.8
NDCG@5	0.77	0.89	15.4

5.3 Sensitivity Analysis of Recommendation Algorithm Parameters

The parameter sensitivity analysis employs a combination of grid search and Bayesian optimization to systematically evaluate the impact of key hyperparameters on model performance. The main parameters of the hybrid recommendation model include latent vector dimension k (32,256), regularization coefficient λ (0.01 to 0.2), learning rate η (0.0001 to 0.01), and the number of attention heads h (4 to 16). Experimental results indicate that when k=64, λ=0.08, η=0.001, and h=8, the model performance is optimal. Increasing the latent vector dimension beyond 128 shows minimal performance improvement but significantly increases training time. The regularization coefficient has a pronounced effect on the model's generalization ability; if too small, it leads to overfitting, while if too large, it results in underfitting. The choice of learning rate requires a balance between convergence speed and stability, and employing a cosine annealing strategy yields better training outcomes. Incremental training validates the model's scalability, showing that performance decreases by no more than 5% when processing a tenfold increase in data volume, demonstrating the algorithm's good scalability. Stability testing of parameters indicates that within the optimal parameter neighborhood, performance fluctuations are controlled within 3%. Figure 5 illustrates the three-dimensional surface graph of parameter sensitivity analysis.

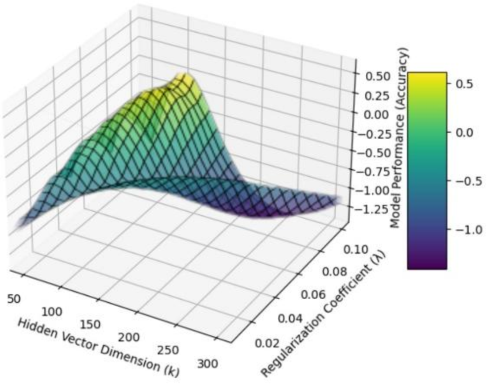

Figure 5. Three-Dimensional Surface Graph of Parameter Sensitivity Analysis

5.4 Evaluation of System Application Effectiveness

The system was deployed in a production environment for a large-scale A/B test over three months, with the experimental group and control group each comprising 50,000 users. Evaluation dimensions include learning outcomes (completion rates, mastery levels), user experience (satisfaction, retention rates), and system performance (response time, resource utilization). Objective data shows that users in the experimental group achieved an average course completion rate increase of 24.3%, knowledge mastery improved by 18.7%, and learning duration increased by 31.2%. User survey feedback indicates that the personalized recommendation system significantly enhances learning efficiency and experience, with a satisfaction score of 4.2 out of 5. Analysis of learning behavior reveals that users in the experimental group had a more rational distribution of learning time, with knowledge coverage improving by 15.6%, and learning paths aligning more closely with cognitive patterns[10]. System performance monitoring indicates that with daily active users around 100,000, the average response time remains below 300 ms, CPU utilization peaks do not exceed 60%, and memory usage stabilizes below 40 GB, confirming the system's good scalability and stability. ROI analysis indicates that after deploying the system, the platform's customer acquisition cost decreased by 23.5%, and user lifetime value increased by 28.7%.

6. Conclusion

This study has designed and implemented a personalized recommendation algorithm framework for an adaptive learning system. Through multidimensional learner modeling, educational data mining, and knowledge graph techniques, a hybrid recommendation model that combines collaborative filtering and knowledge graphs was constructed. The dynamic learning path generation method based on the A* algorithm enables intelligent planning of learning paths. Experimental results show that this framework outperforms traditional methods in recommendation accuracy, learning efficiency, and user experience, achieving a 24.3% increase in course completion rates and an 18.7% improvement in knowledge mastery. Additionally, the system exhibits good scalability and stability, providing effective technical support for personalized learning in online education platforms.

References

[1] Wang C, Li C, Huang H, et al. ASNN-FRR: A traffic-aware neural network for fastest route recommendation[J]. GeoInformatica, 2023, 27(1):39-60.

[2] Zhu Y. Personalized Recommendation of Educational Resource Information Based on Adaptive Genetic Algorithm[J]. International journal of reliability, quality and safety engineering, 2023, 30(2):2250014.1-2250014.25.

[3] Chen J Y, Saeedvand S, Lai I W. Adaptive Learning Path Navigation Based on Knowledge Tracing and Reinforcement Learning[J]. 2023.

[4] Chetana V L, Seetha H. CF-AMVRGO: Collaborative Filtering based Adaptive Moment Variance Reduction Gradient Optimizer for Movie Recommendations[J]. International Journal of Computers & Applications, 2022.

[5] Sivamayilvelan K, Rajasekar E, Vairavasundaram S,et al.Flexible recommendation for optimizing the debt collection process based on customer risk using deep reinforcement learning[J].Expert Systems With Applications, 2024, 256.

[6] Cho Y B. XAI Personalized Recommendation Algorithm Using ViT and K-Means[J].Journal of Electrical Engineering & Technology, 2024, 19(7):4495-4503.

[7] Boughida A, Kouahla M N, Lafifi Y. emoLearnAdapt: A new approach for an emotion-based adaptation in e-learning environments[J]. Education and Information Technologies, 2024, 29(12):15269-15323.

[8] Wang F, Zhang L, Chen X, et al. A personalized self-learning system based on knowledge graph and differential evolution algorithm[J]. Concurrency and computation: practice and experience, 2022(8):34.

[9] Gao Y, Wang L, Li M, et al. Information Recommendation Method of Network Blended Skill Learning Based on Personalized Adaptive of Electric Power Staff[J]. Springer, Singapore, 2022.

[10] Rahmani S, Hosseini S, Zall R, et al. Transfer-based adaptive tree for multimodal sentiment analysis based on user latent aspects[J]. Knowledge-Based Systems, 2023, 261:110219.