Journal of Computer Technology and Software
ISSN:2998-2383
Vol. 3, No. 8, 2024

Value-Aware Recommendation Modeling Through Reinforcement Learning and Offline Interaction Data

Kaili Zhang[1*]
[1]Westcliff University, Irvine, USA
*Corresponding author: Kaili Zhang; zklkaili@gmail.com

Abstract: This paper studies the modeling and optimization of recommendation algorithms driven by reinforcement learning, abstracting the recommendation process into a sequential decision-making task under continuous interaction to maximize long-term cumulative rewards. In this framework, user history and item-side information are encoded as state representations. Recommendation actions are generated by a policy network in the candidate space, while a value network and a critic network are used to estimate future rewards and construct low-variance learning signals, thereby improving the stability and effectiveness of policy updates. To adapt to the training characteristics of offline interaction logs, the method performs bounded and normalized processing on the feedback signal at the reward level, and combines baseline and advantage modeling to mitigate the bias caused by gradient variance and short-sighted decision-making, achieving synergistic optimization of policy learning and value estimation. Experimentally, a unified evaluation setting is constructed on a public dataset. Multi-method comparisons show that the proposed framework consistently outperforms the compared baselines on ranking quality and hit-rate metrics, indicating that it leverages historical interactions more effectively to make robust recommendation decisions. Overall, this paper provides a reinforcement learning recommendation modeling paradigm oriented towards long-term value optimization, offering structured methodological support for the design of trainable recommendation strategies in dynamic preference scenarios.

Keywords: Sequence decision modeling; offline policy learning; value estimation; reward normalization

1. Introduction

With the rapid development of internet platforms and the digital economy, the scale of information has exploded, making it increasingly difficult for users to make effective choices from massive amounts of content[1]. Recommendation systems, as a crucial bridge connecting users and information, play a central role in e-commerce, online media, social networks, and smart services. Traditional recommendation methods, mostly based on collaborative filtering, matrix factorization, or deep representation learning frameworks, have made significant progress in characterizing users' static interests. However, in real-world applications, user interests exhibit dynamic evolution, and user behavior is often influenced by multi-stage interaction and feedback mechanisms. Single-step optimization strategies are insufficient to fully characterize complex decision-making processes. Therefore, it is necessary to re-examine the modeling paradigm of recommendation problems from the perspective of sequential decision-making and long-term reward optimization[2].

Reinforcement learning provides a systematic theoretical framework for solving sequential decision-making problems. Its core lies in continuously updating strategies through environmental interactions to maximize long-term cumulative rewards. In recommendation scenarios, a continuous interactive relationship exists between the platform and users; each recommendation result influences subsequent behavior, forming a dynamic closed-loop system. Modeling the recommendation process as a Markov decision process helps to simultaneously consider immediate feedback and long-term value, thus overcoming the limitations of traditional supervised learning, which relies solely on static predictions based on historical labeled data. This modeling approach effectively alleviates the decline in user satisfaction caused by short-term click-through rate optimization, providing a theoretical foundation for building a sustainable intelligent recommendation mechanism[3, 4].

In complex application environments, recommendation systems not only need to improve personalized matching accuracy but also need to balance multiple objectives such as user experience, diversity constraints, and platform revenue. Reinforcement learning, through policy optimization mechanisms, can achieve dynamic trade-offs under multi-objective constraints and balance the discovery of known preferences and potential interests through exploration and utilization mechanisms[5]. Furthermore, in the context of large-scale data and high-dimensional feature spaces, reinforcement learning combined with deep representation learning can construct end-to-end decision models, thereby maintaining stable performance in nonlinear and high-noise environments. This fusion paradigm lays an important technical foundation for building adaptive, scalable recommendation systems with long-term optimization capabilities[6, 7].

From both theoretical and application perspectives, research on reinforcement learning-driven recommendation algorithms is of great significance. Theoretically, it helps to promote the transformation of recommendation systems from a static

prediction paradigm to a dynamic decision-making paradigm, expanding the application boundaries of intelligent decision-making theory in the field of information services. On a practical level, this can enhance user stickiness and platform value, achieving synergistic growth in user benefits and platform revenue. Furthermore, against the backdrop of the continuous evolution of artificial intelligence technology, exploring effective modeling and optimization mechanisms for reinforcement learning in recommendation systems has a profound impact on building a more intelligent and sustainable information service ecosystem.

2. Datasets and Data Preprocessing

2.1 Dataset

The MovieLens dataset is a classic open-source dataset in the field of recommender systems. It contains structured data such as user rating records for movies, movie metadata, and tag information, which can be used to construct user-item interaction matrices and conduct user preference learning and evaluation. This dataset offers multiple scale versions, including those with millions and tens of millions of ratings, to meet the needs of training and comparing models of different scales. The explicit feedback-based rating values in the dataset can serve as reward signals or state feedback in reinforcement learning environments, supporting policy learning for multi-step decision-making process modeling and long-term reward optimization. The widespread use of the MovieLens dataset provides a standardized experimental benchmark and reproducible data foundation for reinforcement learning-driven recommender algorithm research.

In specific applications, the latest scale version of MovieLens can be used as the research object. This version contains tens of millions of rating records and rich tag applications, covering large-scale user behavior and item attribute information. This data structure can be used to construct the state space, action space, and reward mechanism in a reinforcement learning recommender environment, mapping user historical interaction sequences and simulating the dynamic effects of real-time feedback during online recommendation. Using such open-source datasets not only enables effective evaluation of algorithm performance but also facilitates the comparison and verification of research results in academia and industry, thereby promoting the theoretical development and practical application of reinforcement learning recommendation algorithms.

2.2 Dataset preprocessing

In the data preprocessing stage, the original interaction records are first deduplicated and outlier-removed, with missing scores and invalid user identifiers removed to ensure data integrity and consistency. Then, user behavior sequences are sorted according to timestamps, and user and item identifiers are mapped to continuous integer indices to construct a unified encoding space. When building the reinforcement learning environment, the user's historical interaction sequences are divided according to time sequence to form state representations and candidate action sets, and the score values are normalized to serve as reward signals input to the model. Simultaneously, auxiliary features such as movie categories and tags are one-hot encoded or embedded to enhance the expressive power of state features, thereby providing structured and trainable data input for subsequent reinforcement learning strategy optimization.

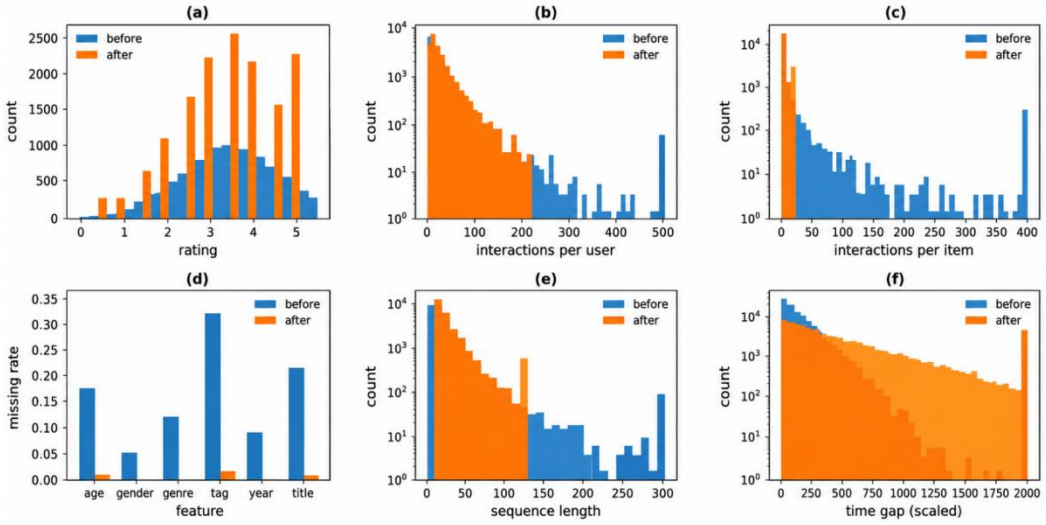

Figure 1. Comparison of experimental results before and after dataset preprocessing

Figure 1 shows the comparison results of six key statistical distributions before and after dataset preprocessing. (a) is the histogram of the rating distribution. After preprocessing, the ratings are mainly concentrated in the legal interval and the overall shape is smoother. Abnormal ratings and noise peaks are effectively suppressed, making the reward signal more stable and reliable. (b) is the distribution of user interaction frequency. Preprocessing significantly reduces the extremely sparse tail by deduplication and filtering of low-activity users, so that most users have more historical interactions to construct state sequences. (c) is the distribution of item interaction frequency. After preprocessing, the proportion of low-frequency items decreases and the long tail is more regular, which helps to alleviate the learning instability caused by too

many unpopular items in the action space. (d) is the distribution of the proportion of discrete side information features. After preprocessing cleans up missing and abnormal encodings, the proportion of each feature value is more consistent, improving the consistency and interpretability of state feature expression. (e) is the distribution of user sequence length. Preprocessing reduces excessively short sequences and makes the main interval more concentrated, thereby enhancing the context information required for sequence decision modeling. (f) is the distribution ofthe time interval. After preprocessing, the ultra-large time interval is compressed and the distribution shows a smoother decay shape, indicating that timestamp anomalies and invalid intervals are corrected. Overall, preprocessing significantly improves the effectiveness, density, and statistical regularity of the data, providing a more stable data foundation for state construction, reward modeling, and long-term benefit optimization in reinforcement learning recommendation.

3. Method

For reinforcement learning driven recommendation decision problems, the core objective is to simultaneously balance immediate feedback and long-term value during the continuous interaction process, thereby avoiding policy bias caused solely by short-term click or rating signals. To this end, the recommendation process is modeled as a sequential decision-making task, enabling the system to form state representations based on user historical behaviors and make action selections over the candidate item set to maximize the cumulative return. The significance of this modeling lies in elevating recommendation from static prediction to an optimizable dynamic control process, where the state is used to carry the evolution trajectory of user preferences, the action corresponds to specific recommended items or recommendation lists, and the environmental feedback characterizes the response intensity of the user to the recommendation results, which in turn drives the iterative updating of the policy to achieve a robust improvement in long term revenue.

$$\mathcal{M} = \langle \mathcal{S} \mathcal{A} P r \gamma \rangle$$

$$J(\pi) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r\left(s_t a_t\right)\right]$$

In the above definition $\mathcal{S}$ and $\mathcal{A}$ represent the state space and action space, respectively $P\left(s_{t+1}|s_t a_t\right)$ describes the environment transition mechanism $r\left(s_t a_t\right)$ gives the reward signal formed by interaction feedback, and $\gamma \in (0,1)$ serves as a discount factor to emphasize long-term return and suppress excessive shortsightedness. To enable the state to retain long-term preferences while reflecting recent interest changes, the user interaction sequence is mapped into learnable representations and organized chronologically as the decision context, thereby enhancing the perception capability of the policy regarding preference drift and phased interests. This paper presents the overall model architecture, as shown in Figure 2.

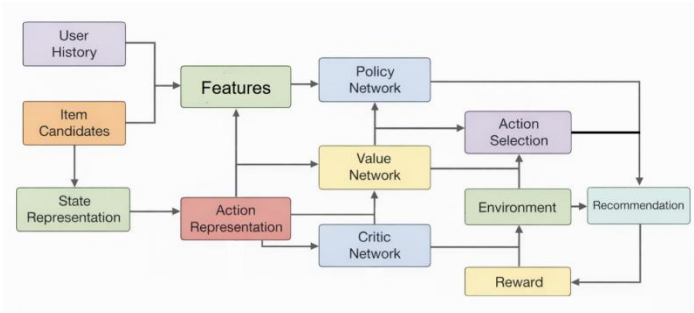

Figure 2. Overall model architecture

Different from the practice of treating recommendation actions as isolated classification choices, policy learning introduces probabilistic decision making to support the balance between exploration and exploitation, thus continuously mining potential interests within uncertain preferences and the long tail item space. The policy outputs the distribution over candidate actions conditioned on the state, and then generates recommendation actions through sampling or maximization selection, allowing the system to stably exploit known preferences while exploring items that may bring higher long term value in a controlled manner.

$$\pi_\theta(a|s) = \frac{\exp\left(f_\theta(sa)\right)}{\sum_{a' \in \mathcal{A}} \exp\left(f_\theta(sa')\right)}$$

$$a_t \sim \pi_\theta\left(\cdot|s_t\right)$$

Here $f_\theta(sa)$ denotes the preference scoring function parameterized by θ which is used to measure the relative priority of action a under state s while $\pi_\theta(a|s)$ normalizes it into a differentiable probability distribution for end-to-end optimization. Based on this formulation, the policy gradient can directly optimize the expected return, avoiding strong assumptions about the environment model while aligning the update direction with long term revenue.

$$\nabla_\theta J\left(\pi_\theta\right) = \mathbb{E}_{\pi_\theta}\left[\nabla_\theta log \pi_\theta\left(a_t|s_t\right) G_t\right]$$

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}$$

To reduce the variance of gradient estimation and improve training stability, a baseline function is integrated into the update signal, making the advantage value semantically represent the gain of a certain action relative to the average level of the current state, thereby focusing the learning emphasis on decisions that truly bring long-term improvements.

$$A_t = G_t - b\left(s_t\right)$$

To characterize the long term value more finely and improve the accuracy of credit assignment, a state value function is introduced to estimate future returns, so that the policy update not only relies on sampled returns but can also utilize learnable value approximations to provide training targets with lower variance. The significance of this design is to bind the improvement of the recommendation policy with long-term revenue prediction, enabling the model to still form a

stable sense of direction in sparse feedback scenarios and achieve more reasonable reward propagation in the multi-step interaction chain.

$$V_{\phi}(s) = \mathbb{E}_{\pi}\left[G_t | s_t = s\right]$$

$$Q_{\psi}(s,a) = \mathbb{E}_{\pi}\left[G_t | s_t = s a_t = a\right]$$

Considering that the advantage value can be given by the difference between the action value and the state value, thereby semantically separating the inherent attractiveness of the state from the additional return brought by the action, this further improves the interpretability and robustness of the policy update.

$$A(s,a) = Q_{\psi}(s,a) - V_{\phi}(s)$$

Furthermore, the temporal difference concept is adopted to construct a self-consistent value learning target, allowing the value function to be effectively updated on finite interaction trajectories and establishing a recursive relationship through the one-step return and the next state estimation.

$$\delta_t = r_t + \gamma V_{\phi}\left(s_{t+1}\right) - V_{\phi}\left(s_t\right)$$

Facing the discrete action space and long tail distribution in practical recommendations, the reward design needs to balance feedback intensity and learning stability; thus, explicit ratings or implicit behavioral signals are mapped into bounded rewards to suppress the interference of outliers on policy updates and improve the consistency of gradient scales. The implication of this processing is to make policy optimization pay more attention to the relative preference structure rather than extreme noise, thereby enhancing the numerical stability of the long-term training process, while being more consistent with the subjectivity and uncertainty of feedback signals in recommendation scenarios.

$$\tilde{r}_t = \frac{r_t - r_{\min}}{r_{\max} - r_{\min}}$$

$$\mathcal{L}_V(\phi) = \mathbb{E}\left[\left(\delta_t\right)^2\right]$$

Under this formulation $\tilde{r}_t$ represents the normalized reward $r_{\min}$ and $r_{\max}$ are the lower and upper bounds of the reward, respectively, and $\mathcal{L}_V(\phi)$ constructs the regression target of the value function centered on the temporal difference error δ_t enabling the value network to continuously and adaptively update on dynamic interaction data. Ultimately, by jointly optimizing policy updates and value learning, the recommender system can achieve a unified modeling from short term feedback to long term satisfaction within the interaction closed loop, thereby providing an interpretable, scalable, and long-term revenue-oriented algorithmic foundation for reinforcement learning driven recommendation decisions.

4. Experimental Results and Analysis

4.1 Experimental setup

The experimental setup employed offline interaction logs to construct a reinforcement learning recommendation environment. User behavior sequences were organized chronologically to ensure causal consistency in the decision-making process. User historical interactions and item-side information were co-encoded as state inputs. The action space consisted of a set of candidate items, and computational overhead was controlled through sampling. The reward signal was provided by normalized feedback intensity to improve training stability. During training, a fixed random seed was used to ensure reproducibility. Optimization employed mini-batch iterations combined with gradient pruning to suppress gradient explosion. Periodic evaluations were used to monitor the convergence trend and policy degradation risk during the training process. The hardware and software platform and key hyperparameter configurations are shown in Table 1.

Table 1. Detailed hyperparameter settings

Item	Configuration
Operating System	Ubuntu 20.04 LTS
CPU	Intel Xeon 32 vCPU
GPU	NVIDIA RTX 3090 24GB
Memory	128 GB
Storage	1 TB SSD
Deep Learning Framework	PyTorch 2.3
Python	3.10
CUDA / cuDNN	CUDA 11.8 / cuDNN 8.2
Batch Size	256
Maximum Sequence Length	50
Number of Candidate Action Samples	200
Hidden Dimension	128
Optimizer	AdamW
Learning Rate	1e-3
Weight Decay	1e-4
Discount Factor	0.99
Gradient Clipping Threshold	1.0
Number of Training Steps	5000
Random Seed	42

4.2 Experimental Results and Analysis

To systematically present the progress of reinforcement learning-driven recommendation algorithms in sequential decision modeling, long-term payoff optimization, and offline log learning, representative reinforcement learning recommendation works in the same research direction were selected as comparison objects, and a comparative analysis was organized under a unified evaluation index system. The comparison method covers typical paradigms such as list recommendation based on value decomposition, offline correction based on policy gradient, interactive recommendation based on actor-critic structure, and offline reinforcement learning constraint learning, to ensure coverage of different decision modeling paths and optimization mechanisms. Table 2 shows the comparison records of the comparative methods and the proposed method on four evaluation indexes.

Table 2. Experimental results compared with other models

Method	NDCG@10	HR@10	Recall@10	MAP@10
Zheng et al.[8]	0.47	0.66	0.51	0.29
Zhao et al.[9]	0.49	0.68	0.53	0.31
Ie et al.[10]	0.51	0.70	0.55	0.33
Chen et al.[11]	0.50	0.69	0.54	0.32
Chen et al.[12]	0.52	0.71	0.56	0.34
Xiao et al.[13]	0.53	0.72	0.57	0.35
Xin et al.[14]	0.54	0.73	0.58	0.36
Ours	0.58	0.77	0.62	0.40

Table 2 shows the performance of different methods on four ranking and hit-related metrics. Our proposed method achieves superior performance on all these metrics, reflecting that the modeled sequential decision-making mechanism can more fully characterize the dynamic evolution of user preferences during the interaction process and balance immediate feedback and long-term value when selecting recommended actions. This result indicates that the joint optimization of policy output and value estimation can provide a more stable utility characterization for the ranking of the recommendation list, thereby improving overall relevance and effective hit capability.

From the perspective of metric connotation, our proposed method maintains a consistent advantage in ranking quality and hit coverage, indicating that state representation is more effective in organizing historical behavioral information, action selection better reflects the user's true preferences in the candidate space, and reward modeling more effectively suppresses noisy feedback. In summary, the long-term reward objective under the reinforcement learning paradigm can reduce the bias caused by short-sighted decision-making, making the recommendation strategy more sustainable and robust in multi-step interaction scenarios. Therefore, our proposed method performs better.

The policy temperature parameter is used to adjust the smoothness of the action distribution, thus affecting the trade-off between exploration intensity and exploitation preference. Different temperature settings change the probability concentration of the policy output, thereby affecting the distribution of sampled and executed recommended actions. To characterize the impact of this hyperparameter on policy behavior, a fixed training process and consistent data processing method were used; only the temperature parameter was changed, and the corresponding performance response was recorded. The experimental results are shown in Figure 3.

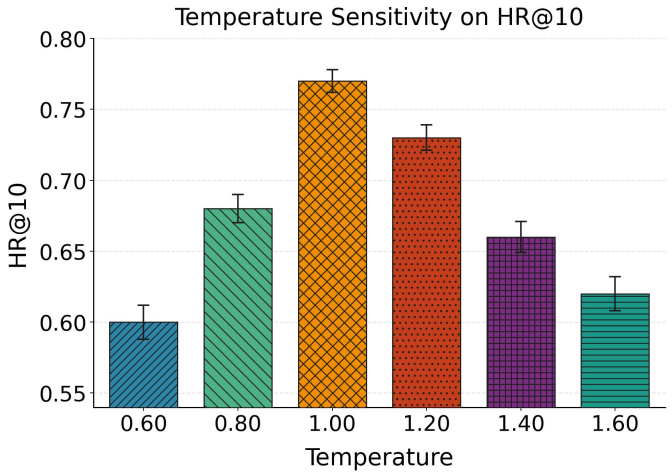

Figure 3. Experimental results on the sensitivity of strategy temperature parameters to HR@10

This sensitivity analysis focuses on the policy temperature parameter to characterize how the smoothness of the policy distribution affects the performance of recommendation decisions. Temperature changes directly impact the concentration of action probabilities, thus affecting the balance between the stability and diversity of candidate item sampling and recommendations. This experiment verifies the behavioral differences in policy output under different exploration intensities at a mechanistic level, providing a basis for subsequent hyperparameter selection.

The results show that the proposed method achieves better performance under moderate temperature settings, indicating that the policy fully utilizes high-value actions while avoiding insufficient exploration due to over-conservatism. When the temperature is too low, the policy distribution becomes too sharp, easily leading to singular recommended action choices and amplifying early biases; when the temperature is too high, the policy distribution becomes too flat, increasing the randomness of action selection and diluting the effective decision signal, making it difficult to stably focus on recommendation paths with better long-term returns.

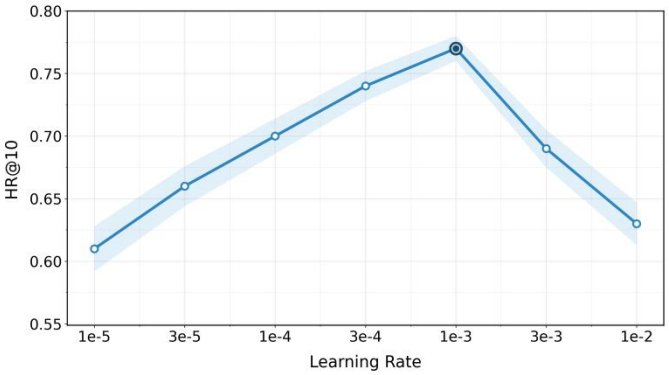

Figure 4. Results of experiments on the sensitivity of learning rate to HR@10

Figure 4 illustrates the variation of HR@10 under different learning rate settings. The results show that a suitable learning rate keeps the policy update magnitude and value estimation error within a more coordinated range, resulting in more stable recommended action selection that is closer to high-value decisions. A reasonable update step size helps to strike a balance between exploration and exploitation, avoiding both excessive conservatism leading to insufficient coverage of effective actions and suppressing the dilution of decision signals caused by excessive randomness. Therefore, our method performs better under this setting.

From the perspective of the policy learning mechanism, when the learning rate deviates from a reasonable range, parameter updates are either too aggressive, causing policy distribution oscillations and amplifying early biases, or too slow, reducing policy improvement efficiency and making it difficult to fully absorb reward feedback, thus affecting the stable formation of hit capability. The confidence bandwidth in the figure reflects the level of fluctuation during training. A tighter fluctuation range is observed near the optimal learning rate, indicating that the update process is more likely to converge to a consistent decision pattern and maintain reliable recommendation output. Therefore, our method performs better.

5. Conclusion

This paper focuses on reinforcement learning-driven recommendation algorithms, emphasizing the view of the recommendation process as a sequential decision-making

problem under continuous interaction. It uses long-term cumulative reward as the core optimization objective, forming a unified modeling framework from state construction to policy learning and value estimation. This approach, at the methodological level, propels recommendation systems from static preference fitting to sustainable dynamic decision-making, enabling the model to maintain stable decision-making capabilities under the combined influence of real-world factors such as user interest evolution, feedback uncertainty, and the long tail of the candidate space. By co-incorporating probabilistic policy output and value function estimation into the optimization process, the recommendation strategy can not only more effectively utilize the preference structure inherent in historical behavior but also achieve a more reasonable trade-off between exploration and utilization, thereby improving the reliability and interpretability of the overall recommendation behavior.

From an application perspective, the reinforcement learning paradigm provides recommendation systems with an optimization path closer to real-world business loops. It can coordinate user experience and platform goals within the same decision-making perspective, avoiding policy bias and satisfaction loss caused by simply pursuing short-term feedback. This framework applies to high-frequency interaction scenarios such as content distribution, e-commerce, short videos, and social information feeds, and can be naturally extended to recommendation decisions under multi-objective constraints, such as considering relevance, diversity, and novelty. Because the strategy outputs a decision distribution conditioned on the state, its behavioral logic can be more closely integrated with business rules, risk control, and compliance constraints. This provides a methodological foundation for controllable and auditable intelligent recommendations and has practical significance for improving the sustainability of the information service ecosystem and user trust.

Methodologically, the Markov decision process modeling adopted in this study provides a clear structured abstraction for recommendation tasks, enabling a more explicit expression of the causal relationship between states, actions, and rewards. This facilitates scalable design under more complex feedback systems. The introduction of value estimation and advantage modeling allows policy updates to more directly align with long-term gains and maintain more stable learning signals in sparse or noisy feedback scenarios. Reward normalization and stable training mechanisms further enhance the robustness required for practical deployment, making the model more adaptable to changes in user behavior distribution and data quality fluctuations. Overall, this research provides a reusable modeling paradigm and optimization ideas for the application of reinforcement learning in recommendation systems and lays the foundation for further deep integration of reinforcement learning with representation learning, constraint optimization, and other directions.

Future work can start from more realistic online interactive environments and richer feedback signals to further enhance the strategy's ability to characterize complex user intentions and multi-stage behavioral chains, and explore more generalizable offline-to-online migration mechanisms to reduce deployment costs and risks. In actual business, recommendation decisions are often influenced by multiple objectives, constraints, and agent factors. Future research can consider introducing explicit constraint modeling and controllable strategy generation to enable recommendation systems to maintain long-term value optimization capabilities while meeting compliance and fairness requirements. Meanwhile, with the development of large-scale models and intelligent agent technologies, combining sequential decision-making with higher-level intent understanding, knowledge enhancement, and interactive feedback is expected to improve the recommendation system's adaptability to semantic preferences and contextual needs. For industry applications, the reinforcement learning recommendation paradigm proposed in this research is expected to promote more robust and sustainable information distribution mechanisms, bringing broader practical impacts to the digital content ecosystem, personalized services, and intelligent business systems.

Recent studies further indicate that reinforcement learning recommendation can be extended through more reliable representation, infrastructure-aware deployment, and semantic matching mechanisms. Self-supervised unified representation learning with structural dependency modeling provides a useful perspective for capturing early risk signals in distributed environments, which can strengthen the robustness of recommendation services operating on large-scale platforms [15]. Graph neural network-driven anomaly detection for multi-tenant cloud security and lightweight anomaly detection in distributed edge-cloud systems show that deployed intelligent services require continuous monitoring of hidden state changes and operational risks [16], [17]. Reinforcement learning-based task scheduling under energy budget constraints connects sequential decision optimization with resource-aware execution, offering a system-level reference for recommendation policies that must balance effectiveness, latency, and cost [18]. Masked language modeling for semantic matching and text robustness is also relevant to recommendation scenarios that rely on textual items, user queries, and noisy behavioral descriptions [19]. Dynamic resource load forecasting and structure-aware multi-scale behavioral learning further suggest that representation learning can capture time-varying load, observability, and behavioral dependencies, thereby supporting more adaptive and stable recommendation deployment under complex operating conditions [20], [21].

References

[1] Gao C, Huang K, Chen J, et al. Alleviating matthew effect of offline reinforcement learning in interactive recommendation[C]//Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval. 2023: 238-248.

[2] Wu J, Xie Z, Yu T, et al. Dynamics-aware adaptation for reinforcement learning based cross-domain interactive recommendation[C]//Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval. 2022: 290-300.

[3] Dai Y, Sun W. Representation Learning in Low-rank Slate-based Recommender Systems[J]. arXiv preprint arXiv:2309.08622, 2023.

[4] Li M, Huang J, de Rijke M. Repetition and exploration in offline reinforcement learning-based recommendations[C]//4th Workshop on Deep Reinforcement Learning for Information Retrieval at CIKM. 2023.

[5] Deffayet R, Thonet T, Renders J M, et al. Offline evaluation for reinforcement learning-based recommendation: a critical issue and some alternatives[C]//Acm sigir forum. New York, NY, USA: ACM, 2023, 56(2): 1-14.

[6] Wang J, Karatzoglou A, Arapakis I, et al. Reinforcement learning-based recommender systems with large language models for state reward and action modeling[C]//Proceedings of the 47th International ACM SIGIR conference on research and development in information retrieval. 2024: 375-385.

[7] Zhang Y, Qiu R, Liu J, et al. ROLeR: Effective Reward Shaping in Offline Reinforcement Learning for Recommender Systems[C]//Proceedings of the 33rd ACM International Conference on Information and Knowledge Management. 2024: 3283-3292.

[8] Zheng G, Zhang F, Zheng Z, et al. DRN: A deep reinforcement learning framework for news recommendation[C]//Proceedings of the 2018 world wide web conference. 2018: 167-176.

[9] Zhao X, Xia L, Zhang L, et al. Deep reinforcement learning for page-wise recommendations[C]//Proceedings of the 12th ACM conference on recommender systems. 2018: 95-103.

[10] Ie E, Jain V, Wang J, et al. SlateQ: A Tractable Decomposition for Reinforcement Learning with Recommendation Sets[C]//IJCAI. 2019, 19: 2592-2599.

[11] Chen M, Beutel A, Covington P, et al. Top-k off-policy correction for a REINFORCE recommender system[C]//Proceedings of the twelfth ACM international conference on web search and data mining. 2019: 456-464.

[12] Chen X, Huang C, Yao L, et al. Knowledge-guided deep reinforcement learning for interactive recommendation[C]//2020 International Joint Conference on Neural Networks (IJCNN). IEEE, 2020: 1-8.

[13] Xiao T, Wang D. A general offline reinforcement learning framework for interactive recommendation[C]//Proceedings of the AAAI Conference on Artificial Intelligence. 2021, 35(5): 4512-4520.

[14] Xin X, Karatzoglou A, Arapakis I, et al. Supervised advantage actor-critic for recommender systems[C]//Proceedings of the fifteenth ACM international conference on web search and data mining. 2022: 1186-1196.

[15] T. Ying and C. Ding, "Self-Supervised Unified Representation Learning with Structural Dependency Modeling for Early Risk Detection in Distributed Systems," 2024.

[16] S. Shawulieti, "Enhancing Cloud Security with Graph Neural Network-Driven Anomaly Detection in Multi-Tenant Environments," 2024.

[17] S. Xu, "Efficient Anomaly Detection in Distributed Edge-Cloud Systems Using Lightweight Modeling," 2024.

[18] J. Sun, "Reinforcement Learning-Based Task Scheduling Under Energy Budget Constraints in Edge-Cloud Systems," 2024.

[19] Z. Bian and X. Su, "Enhancing Semantic Matching and Text Robustness through Masked Language Modeling," 2024.

[20] T. Xia, "Dynamic Resource Load Forecasting in Cloud Computing: A Representation Learning Perspective," 2024.

[21] Y. Zhou, "Structure-Aware Multi-Scale Behavioral Learning for Distributed System Observability and Anomaly Identification," 2024.