ISSN 2581 – 3781

AJCSE
ASIAN JOURNAL OF COMPUTER SCIENCE ENGINEERING

Available Online at www.ajcse.info
Asian Journal of Computer Science Engineering 2025;10(3):1-10

REVIEW ARTICLE

Reinforcement Learning Techniques to Continuously Adapt and Optimize Recommender Systems Based on User Interaction Patterns

Jvalant Kumar Kanaiyalal Patel

Department of Computer Application, Shri Manilal Kadakia College of Commerce, Management, Science and Computer Studies, Ankleshwar, Gujarat, India

Received: 15-05-2025; Revised: 30-06-2025; Accepted: 12-07-2025

ABSTRACT

Reinforcement learning (RL) has emerged as a powerful approach in recommender systems, modeling user interactions as sequential decision-making processes to deliver adaptive, personalized, and context-aware recommendations. Unlike traditional methods that focus on short-term accuracy, RL emphasizes long-term user engagement by dynamically responding to evolving behaviors and preferences. This paper systematically reviews RL-based recommender frameworks, including value-based, policy-based, actor–critic, and hybrid approaches, as well as emerging trends such as explainable RL, fairness-aware design, and privacy-preserving mechanisms. Multi-dimensional evaluation metrics, including diversity, novelty, and serendipity, are discussed, alongside integration strategies combining RL with collaborative and content-based filtering for enhanced scalability and robustness. Although there has been significant progress, problems of data sparsity, cold-start situations, computational issues, and interpretability still exist. The review gathers existing research findings that illuminate the limitations and identifies new research avenues that can be used to develop user-friendly scalable, and transparent RL-based recommender systems in future applications. The systems hold the promise of enhancing user satisfaction and engagement greatly across the digital platforms, creating a useful advantage to online retailing, streaming services, and social media. Further ongoing innovation in RL approaches is needed to satisfy the increasing requirements of smart, flexible recommendation systems.

Keywords: Deep q-learning, fairness-aware recommendation, hybrid recommendation models, privacy-preserving frameworks, recommender systems, reinforcement learning, sequential decision-making

INTRODUCTION

Recommender systems have become cornerstones in delivering personalized experiences to marketplaces, streaming entertainment and video content, online education, and social media platforms in the data-intensive digital ecosystems age.[1] Traditional methods of making recommendations, including collaborative filtering (CF), content-based filtering (CBF), and combinations of the two, have shown themselves to work well in some settings. Yet, they have difficulties in adapting to the topic-specific and dynamically changing user preferences.[2] Such techniques are mainly based on static historical information and cannot capture the sequential connections, respond to circumstances, and maximize long-term user involvement. These fixed models have a serious problem, especially when responding to the dynamism of user preferences, item stocks, and changing contextual considerations.[3] Their disability to connect with real-time insights usually leaves recommendations outdated or less relevant, which contains user engagement and satisfaction barriers to a great extent.

Reinforcement learning (RL) presents a strong answer to these difficulties by posing recommendation tasks as an issue in sequential decision-making.[4] In RL-based systems, interactions between the user and the system are represented as a Markov decision process (MDP), and the recommendation agent can thus update its policy many times, trading off exploration (adding

***Corresponding Author:**
Jvalant Kumar Kanaiyalal Patel
E-mail: jvalant007@gmail.com

new items) and exploitation (utilizing known preferences).[5] In contrast to the static ones, RL adjusts its strategy as the real-time feedback reflects accumulated rewards based on the overall user satisfaction, with regard to a combination of clicks and ratings, but not to each click or rating separately. RL also facilitates dynamic personalization by continually updating policies on the fly according to interaction feedback, and it has been used to optimize not only immediate engagement, but long-term user satisfaction as well in recommender systems.

A recommender system has the primary role of linking a user with a relevant content in such areas as apps, games, e-commerce, music, videos, and social media. It emphasizes how individualized recommendations arise as a result of user choices and interactions, as illustrated in Figure 1. This review emphasizes ways RL and DRL can improve recommender systems by adapting to changing user preferences, maximizing long-term rewards, and adopting individualized user experiences on very large platforms.

Structure of this Paper

The paper is divided into the following sections: Section II presents the principles of RL in the recommendation models. Section III describes the reinforcement techniques in recommendations. In Section IV, it is explained how recommender systems are continuously updated and improved in the user interface. Section V is a discussion of related literature and identification of gaps in the research, and finally, Section VI is the conclusion and future directions of the research done.

FUNDAMENTALS OF RL IN RECOMMENDER SYSTEMS

RL brings a strong recommendation paradigm to suggest that recommendation is a sequential decision-making action. As opposed to conventional methods that make use of unchanging profiles or sets of existing data, RL allows systems to go through the trial and error method in a continuous way, as well as evolving and adapting to the user based on preference. Within this context, the recommendation engine is the actor, the user and the surrounding context is the state, action suggests the recommendation, and the user promotes feedback.[6] This form of trial-and-error learning resembles human and animal learning in that reward cues are acquired using past experiences to develop the best behavioral strategies.

Besides recommender systems, RL has been shown to be flexible in many fields of science and engineering, with efficient solutions to a complex sequential decision problem with little or no model of the system available.[7] A powerful and flexible method for adaptive, personalized, and context-aware recommendation systems, modern

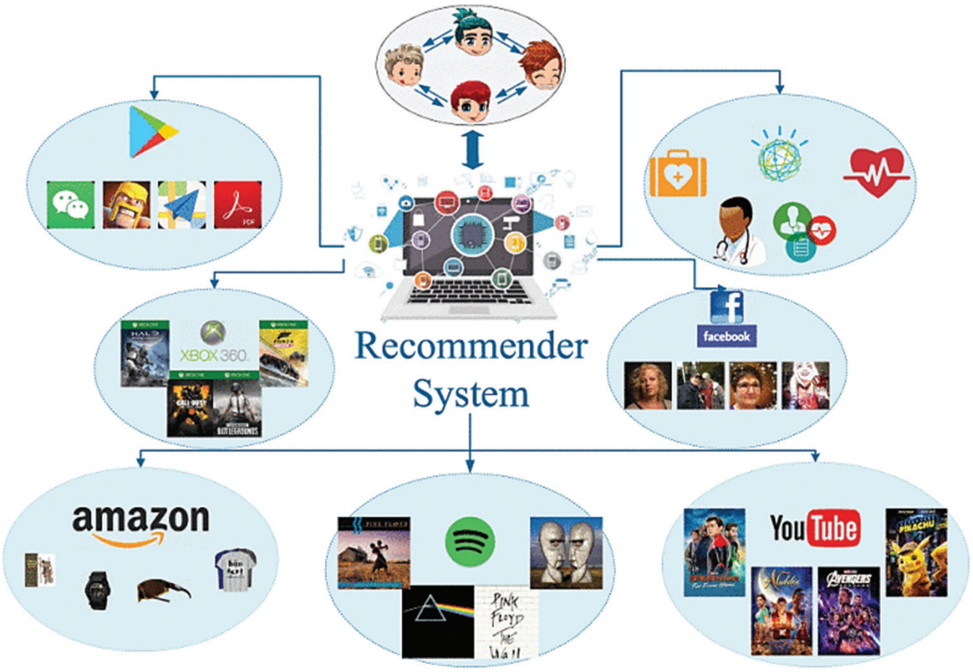

Figure 1: Digital system role in the ecosystem

RL is based on three main areas of study: learning psychology, optimum control through dynamic programming, and temporal difference learning.

Classification of Recommender Systems

Hybrid approaches, CBF, and CF are the three primary varieties of recommender systems, distinguished by the approach taken to generate customized recommendations.[8] Each category adopts a distinct mechanism for inferring user preferences and exhibits unique strengths and limitations in terms of scalability, adaptability, and robustness, as given below:

CF techniques

The foundational premise of CF is that individuals who have consistently displayed similar behavior are likely to continue to do so in the future. Another approach to categorize CF methods is by whether they rely on memory or models. Through the use of similarity metrics such as Pearson correlation and cosine similarity, the memory-based CF directly ascertains the degree of similarity between users or objects.

CBF approaches

CBF produces recommendations using the characteristics of items and compares those items to a user's known preferences. Taking an instance of movie recommendation, CBF can make use of genre, actors, and directors to locate similar movies to those that a user has rated highly in the past. Similarity measures commonly used in this approach are cosine similarity or TF IDF in text-based domains. The strength of CBF is that it does not rely on the data of other users, so it at least partly solves the cold-start problem.

Hybrid recommendation strategies

A hybrid strategy takes advantage of the best features of both content-based and collaborative tactics while avoiding the drawbacks of each. Such strategies can also combine models at various levels, for example, combining the predictions of the two CF and CBF models, or using a different approach when the necessary information is missing, or using other methods such as demographic filtering or knowledge-based suggestions. The use of hybrid techniques is followed in commercial applications mainly due to their increased diversity of recommendations, better solutions to the cold-start situation, and increased prediction accuracy.

Core Components of RL in Recommendations

The different parts of RL, states, actions, environments, agents, and rewards, give us a new way to think about how to make suggestions work better. Any time a user interacts with the recommendation platform; RL-based recommendation systems see it as a series of decisions. In contrast to the "state" that prominently displays the user's current situation, the "actions" indicate the recommendations made to the user. The agent determines the recommended features of the system depending on the user's states and previous interactions, while the environment records the recommendation system [Figure 2]. As a means of providing feedback on the usefulness and quality of an action, rewards are associated with it.

This method enables recommendation systems to evolve with time, bringing in user input and interactions to consistently deliver more relevant and engaging suggestions. Among the numerous potential uses of RL are the exploration of novel items, the optimization of long-term user happiness, and the dynamic adaptation to changing user preferences.

RL TECHNIQUES FOR RECOMMENDATIONS

RL is a type of interactive recommendation system that thinks of the job as a time-series decision-making task. This way, systems can see how to

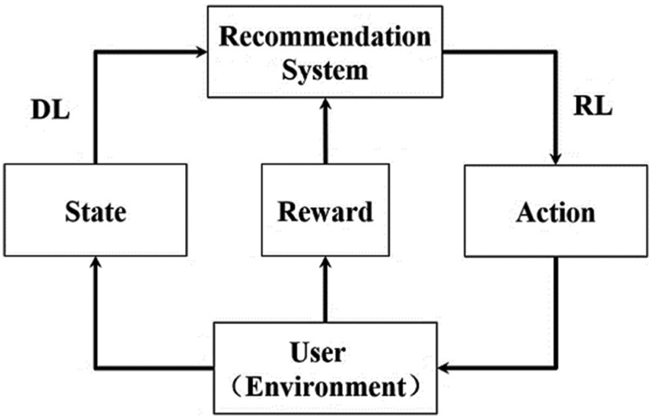

Figure 2: Components of reinforcement learning in recommendation systems

adapt to users' changing preferences by letting them interact and giving them rewards.[9] In contrast to stationary strategies, RL-based strategies keep learning every time the behavior changes, and deep RL (DRL) strategies, such as value-based models, policy-based models, actor-critic models, and model-based RL (MBRL), further allow greater scale and flexibility.[10] Other RL techniques, such as deep Q-networks (DQN), policy gradient, and actor-critic, are techniques that broaden the capacity of RLs to represent high dimensionality in state space, and embody the latent factors of behavioral patterns.[11]

Three main types of recommender system RL methods exist: value-based, policy-based, and hybrid actor-critic. These types differ in their learning strategies and adaptability to user interaction patterns, are given below [Figure 3].

Value-based RL

The aim of value-based RL algorithms is to approximate the expected total reward of a state or state-action pair by learning a value function.[12] Agents can maximize these value estimates indirectly to produce optimal policies. Its mathematical rigor, convergence guarantees, and applicability to decision-making, game-playing, and recommendation-system tasks make these methods popular, and they work well in places with discrete action spaces.

- Q-learning: Q-learning is a value-based RL method that enables an agent to learn the expected reward of actions in given states. By constantly revising the knowledge of past experiences, it improves its course of action-selection with time. Q-learning can be used in recommendation systems as a gainful utilization of user interaction history and future reward depends on how well it has been improved through learning experiences to enhance personalization and make it effective.

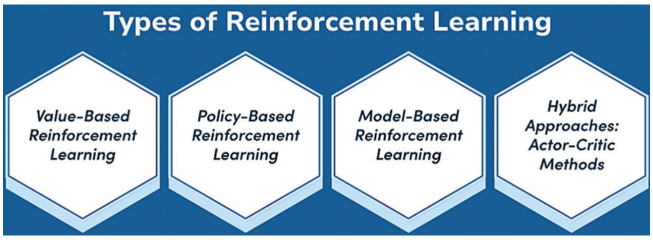
Types of Reinforcement Learning
Value-Based Reinforcement Learning
Policy-Based Reinforcement Learning
Model-Based Reinforcement Learning
Hybrid Approaches: Actor-Critic Methods

Figure 3: Reinforcement learning types in recommendation systems

- DQN: DQNs can learn in high dimensions of state and action spaces based on a neural-network approximation of the Q-function. They are used in recommendation systems, for example, video platforms, E-commerce websites, and news portals as a dynamic adjustment of the recommendation according to user behavior and interests over time to increase relevancy and engagement.

Policy-based RL

Policy-based approaches represent a strategy of directly mapping states to actions and learning optimal policy parameters so that the expected rewards over time are maximized. This allows an easier convergence of action spaces of continuous or high dimensions. In recommendation systems, they come up with suggestions based on the patterns of interaction of the user, context-specific information, and long-term involvement nuclei.[13] The policy-driven approaches also minimize the errors since the organizational rules become directly integrated in the decision process, which makes them especially useful in domains where strict requirements are applied (healthcare, finance, government), where real-time checks on compliance may be ensured.

Actor-Critic RL

Actor-Critic algorithms bridge the gap between policy- and value-based approaches to RL.[14] There are two parts to this: the actor and the critic. The actor makes policy-based action choices, and the critic estimates value functions. Unlike other traditional policy gradient methods[15] which can be extremely variable the Actor-Critic technique provides feedback to the actor training, thus stabilizing actor training. They work in discrete and continuous action spaces; hence, they are good at high-dimensional state-action spaces. The Actor-Critic schemes have been effectively used in robot control, financial decision-making, and recommendation engines, where adaptation and ongoing learning are paramount.

MBRL

A predictive model of the dynamics of the environment is used in MBRL to plan and simulate future encounters. Within recommenders, MBRL leverages the past user-item interactions to profile the behavior of the users and predict their future likes, with the ability to adapt and make smarter and less data-intensive suggestions.[16] MBRL builds an internal model that can be used to simulate user response and reward signal, whereas the model-free approaches directly learn a policy over the interaction data. The simulation potential enables the system to design several recommendation plans before the actual use, saving exploration cost and greatly improving sample efficiency. Model predictive control and Monte Carlo Tree Search are decision-making strategies that should be applied in such contexts.

CONTINUOUS ADAPTATION AND OPTIMIZATION IN RECOMMENDER SYSTEMS THROUGH USER INTERACTION

Recommender systems operate in dynamic settings; user preferences, item availability, and the circumstances all change. To stay effective, such systems must be dynamic and keep upgrading themselves as they continuously change their models to correspond to the dynamics of the user behaviors and the new material. Simultaneously, they are required to maximize their recommendations and find a balance between short-term goals, for example, instant engagement and clicks, and long-term ones, for example, user satisfaction, loyalty, and retention. This constant adaptation to new situations and a strategic approach to optimization make recommendations relevant, individualized, and consistent with user needs and business goals.

Continuous Adaptation

It can be understood as the control of a system that varies dynamically to adapt to any emerging interactions and preferences among its users.[17] It helps to keep recommendations fresh and current by engaging new data and opinions on-the-fly, such as:

Adaptive user preference modeling

The problem with static profiles is that it may become obsolete in dynamic digital scenarios. This is overcome through a real-time adaptive system that identifies past trends, seasonality, situational highs, and recognizes repeat behavior paths to better tailor suggestions within DARS.

Explainable adaptive learning (EAL) module

Improved openness and trust among users are outcomes of the EAL module’s capacity to make recommendation results interpretable. It lays forth the groundwork for the attention mechanisms, analyzes the value of features, and uses counterfactual explanations to highlight the specific reasons for recommending certain content.[18] To close the knowledge gap between model decisions and user understanding, sentiment-based explanations provide greater clarity on how emotional signals impact preference modifications.

Personalized recommendation and output

The aim of a context-aware ranking algorithm is to prioritize material based on user-specific and real-time contextual elements. By implementing diversity and novelty control, the system takes measures to prevent repetition and keep users engaged. This is done by making sure that recommendations contain both familiar and unexpected content. Through constant monitoring of these key performance indicators, performance metrics tracking allows for the A/B testing and dynamic fine-tuning of recommendation techniques.

User evaluation and statistical validation

Digital news platforms, multimedia streaming, and online shopping were the three real-world areas where participants interacted with the Content Delivery Network (CDN). After interacting with the system, users were asked to fill out a standardized survey using 5-point Likert scales to provide input on how easy it was to understand. The following were included as evaluation metrics: user confidence in the system, usefulness, clarity of explanations, and perceived correctness of recommendations.

Optimization Strategies

Optimization strategies for recommender systems aim to enhance their performance by improving accuracy, user satisfaction, and business goals. These strategies can be grouped into model-based, data-centric, and evaluation-driven approaches:

- Evaluation metrics: Accuracy, precision, and recall were used to evaluate the success of the recommendations. Other metrics that were considered to capture relevance and user satisfaction included CTR, conversion rate, dwell duration, novelty, and variety.
- Balancing short-term versus long-term rewards: Optimization must tradeoff between immediate engagement (clicks, purchases) and sustained outcomes such as user retention, loyalty, and trust.
- Offline simulation and testing: Simulation environments allow safe evaluation of new algorithms before deployment, reducing risk.
- A/B testing in recommender systems: The purpose of A/B testing is to evaluate two different implementations of a recommendation algorithm or strategy and find out which one yields better results according to established criteria. Separating users into two groups, one that uses the original system (version A) and the other that uses the modified system (version B), is the idea behind this strategy.

Agent–Environment Interaction Framework

The agent–environment interaction constitutes the theoretical basis of RL and presents a methodological means of modeling decision-making in recommender systems. In this paradigm, the recommendation engine plays the role of an agent that is tasked to offer actionable recommendations to those people regarding products, movies, or articles, given the condition of the environment.[19] The environment comprises external conditions that may influence the decision-making of the agent, i.e., the content repository and user population, the context such as time of day, type of device, or spot location. The agent can sense the environment at every time step with a set of observed features that represent the state, make an action, and is rewarded in terms of feedback. This reward can be either explicit, such as user ratings or purchase confirmations, or implicit, such as clicks, dwell time, or scrolling behavior. Here are the highlights of the RL-based recommender systems framework:

- The agent–environment interaction framework is particularly powerful in recommender systems as it explicitly models the time and sequence of user–system interactions.
- Unlike supervised learning, which relies on static training data, RL adapts to non-stationary environments where user preferences evolve.
- The framework supports delayed effects, capturing long-term user engagement where the impact of recommendations may only emerge after several interactions.
- It enables a balance between exploration (testing novel or uncertain recommendations) and exploitation (leveraging known preferences).
- By simulating dynamic feedback loops, the framework facilitates adaptive and personalized recommendation strategies in complex real-world environments.

Advantages of RL for Dynamic and Continuous Personalization

RL transforms personalization by enabling recommender systems to adapt continuously as user interests evolve in real-time. Unlike models trained on fixed historical data, RL-based recommenders learn dynamically from continuous feedback, optimizing recommendations in response to each user action. This adaptive capability ensures relevance even in non-stationary environments where preferences, trends, and contextual factors change rapidly. In addition, RL is goal-oriented, meaning that it prioritizes the enjoyment of users over short-term gains from individual interactions. This makes RL particularly valuable in e-commerce, streaming services, and online learning, where the objective is sustaining user loyalty rather than individual clicks or purchases, is includes:

- Continuous Adaptation: Instantly adapts suggestions to new information and user activity.
- Long-term optimization: Balances short-term engagement with strategies that promote sustained satisfaction and retention.
- Exploration–exploitation balance: Introduces novel content while leveraging known preferences to maintain accuracy.
- Context-aware personalization: Considers context, including time, place, and device kind, to provide personalized suggestions.

User Interaction in Recommender Systems

User interaction forms the foundation of modern recommender systems, enabling personalized content delivery by capturing how individuals engage with digital environments. By analyzing the nature, frequency, and depth of interactions, systems can predict future actions and build detailed user preference profiles.[20] Interaction data allow recommendation models to evolve from static profiles to dynamic, context-sensitive suggestions that adapt to changing interests. Effective recommendations depend on accurately modeling user interactions and incorporating feedback to enhance personalization. In RL-based systems, feedback, either implicit or explicit, guides the agent in learning preferences and refining its recommendation policy. Figure 4 illustrates an RL-based recommender system, where user interactions generate states, actions, and rewards, allowing the policy network to balance exploration and exploitation for personalized recommendations.

Explicit Feedback

- Explicit feedback: Direct and deliberate input from users. Examples: Numeric ratings (e.g., 1–5 stars), likes/dislikes, written reviews.
- Advantages: Highly informative, easy to interpret, effective for supervised and RL (e.g., reward shaping).[21]
- Limitations: Often scarce due to user reluctance to provide feedback, leading to cold-start and data sparsity challenges.
- Derived from passive user behavior rather than deliberate input. Examples: Clicks, page views, scrolling patterns, dwell time, and purchase history.[22]
- Advantages: Abundant, continuously generated, provides rich behavioral signals.
- Limitations: Noisy and ambiguous (e.g., clicking out of curiosity or leaving a page due to external distraction rather than dissatisfaction).

Practical Consideration

Effective recommender systems typically integrate both explicit and implicit feedback to strike a balance between interpretability and data availability [Figure 5]. While implicit feedback offers scalability through its continuous and abundant nature, explicit feedback enhances accuracy and interpretability by providing clear signals of user preferences.

LITERATURE REVIEW

Table 1 provides a synopsis of the research on recommender system techniques based on RL, outlining their main points, limits, important findings, and potential future directions; this table is used as a basis for both the executive summary and the rest of the analysis.

Kalideen and Yağli (2025) provide the primary machine learning algorithms utilized by recommendation systems, including hybrid techniques, CBF, and CF. New developments, particularly in the fields of deep learning and RL, have substantially enhanced the capabilities of these systems. Methods such as Neural CF and autoencoders are employed to improve scalability and record intricate user-item interactions. Meanwhile, RL maximizes engagement over the long run by allowing for dynamic adaptation in response to real-time user feedback. The study delves into the practical applications of recommender systems across many sectors, with a particular emphasis on their value in e-commerce, entertainment, and education. Specifically, it examines how these systems aid in

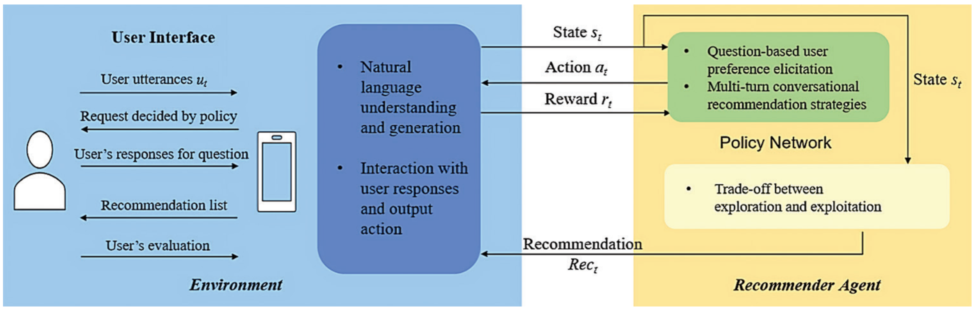
User Interface
User utterances u_t
Request decided by policy
User's responses for question
Recommendation list
User's evaluation
Environment
Natural language understanding and generation
Interaction with user responses and output action
State s_t
Action a_t
Reward r_t
Question-based user preference elicitation
Multi-turn conversational recommendation strategies
State s_t
Policy Network
Trade-off between exploration and exploitation
Recommendation Rec_t
Recommender Agent

Figure 4: User interface with the recommender system

Table 1: Research summary table of reinforcement learning techniques in recommender systems

Reference	Area of focus	Approches	Limitations	Key findings	Future scope
Kalideen *et al*. (2025)	ML algorithms in recommender systems	Content-based, collaborative, hybrid, deep learning, RL	High computational cost, cold-start issue, domain generalization limits	Deep learning improves scalability; RL enables dynamic adaptation and long-term engagement; strong applications in e-commerce, entertainment, education	Use of explainable AI, hybrid/generative models, cross-domain adaptation, fairness and privacy considerations
Boka *et al.* (2024)	Sequential recommendation systems using interaction history	Interaction history, sequential models, evaluation methodologies	Limited discussion on implementation challenges in large-scale systems	Categorized approaches by principle; reviewed evaluation methods; highlighted diverse applications	Identifies open challenges and proposes future research directions in sequential modeling
Liu *et al.* (2023)	Deep RL with enhanced item embeddings using user reviews	REDRL, text review embeddings, MDP, DRL, multi-head self-attention, HIN meta-paths	Increased model complexity; relies on the quality of review data	Introduced review-enhanced DRL with self-attention and meta-path HIN for accurate modeling	Suggests integrating structured data and extending semantic filtering for improved personalization
Lin *et al.* (2023)	Reinforcement learning applied in various RS scenarios: interactive, conversational, sequential, explainable	Interactive, conversational, sequential, explainable RL approaches	Some areas (e.g., real-time feedback mechanisms) less explored; lack empirical comparisons	Summarizes RL use in four key RS types; identifies major challenges and solutions	Emphasizes the development of scalable, real-time, and privacy-preserving RL-based RS
Wu *et al.* (2022)	Use of GNNs in recommender systems across different data types	Taxonomy of GNN models, graph representation learning	Computationally intensive, complex model training	Presents taxonomy based on task and data; addresses how challenges are tackled	Discusses future development of efficient GNNs and integration with other learning paradigms
Salau *et al.* (2022)	Recommender systems in e-learning using deep learning and context-aware approaches	Deep learning, context-aware, hybrid versus traditional methods	Focused mainly on existing studies; lacks experimental insights	Identifies deep learning and context-aware methods as superior to traditional ones	Suggests incorporating more personalized, adaptive, and hybrid approaches in e-learning RS

GNNs: Graph neural networks, RL: Reinforcement learning, MDP: Markov decision process

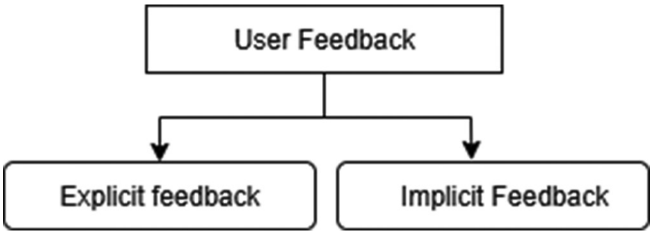

Figure 5: User feedback types in recommender system implicit feedback

product discovery and sales, personalize content suggestions to keep users engaged, and provide individualized learning tools.[23]

Boka *et al.* (2024) delve into the ways in which these systems use user interaction data to provide more tailored suggestions. They include a synopsis of the evaluation, research plans for the future, and procedures used by sequential recommendation systems. They categorize existing approaches according to their guiding principles and evaluate how well they work in various fields. Together with that, describe the possibilities and threats that sequential recommendation systems face. When it comes to data mining and machine learning, recommender systems are formidable tools. In the past, these algorithms were only able to forecast one kind of interaction, such a user's rating of an item.[24]

Liu *et al.* (2023) present REDRL, an approach to interactive recommendation that combines DRL with review upgrades. And can get embedding representations that are enriched by item reviews by using text reviews with a pretrained review representation model. Once have formalized the recommendation problem as an MDP, we can apply DRL to model the interactive suggestion. They introduce a multi-head self-attention method to mimic user choice, which has been neglected in earlier attempts due to the fact that distinct elements in the sequence behavior are considered equally important. Afterward, they delicately combine the meta-paths in heterogeneous information networks (HIN) with the semantic structure information in the user-item bipartite graph to dynamically filter out irrelevant items and acquire candidate items.[25]

Lin *et al.* (2023) provide a comprehensive review, comparisons, and summaries of four typical RL recommendation scenarios: interactive, conversational, sequential, and explainable. In addition, thoroughly review the issue and any applicable remedies using the literature that is already available. Finally, they point out some possible future research options within the context of recognizer systems' limits and outstanding challenges. In many practical contexts, recommender systems have proven invaluable in guiding users to relevant content. In particular, the interactive and autonomous learning capabilities of recommender systems based on RL have made them a hot issue in recent years for academic inquiry.[26]

Wu *et al.* (2022) present that the purpose of this article is to offer a thorough overview of current research on recommender systems that are based on graph neural networks (GNNs). In particular, they classify GNN-based recommendation models based on the kinds and recommendation tasks that are utilized. Furthermore, go over the ways in which previous research in this area has dealt with the difficulties of using GNN on various kinds of data. In addition, provide fresh viewpoints on how this area progresses. Deriving accurate user and item representations from interactions and contextual data, if any, is the main challenge in recommender systems. Due to the graph-structured nature of most recommender system data and GNN's inherent advantages in graph representation learning, GNN techniques have seen increased application in this area as of late.[27]

Salau *et al.* (2022) demonstrated that this survey significantly advances the area of e-learning RSs by surveying existing literature on the topic and offering a variety of suggestions for future e-learning based on both conventional and unconventional recommendation strategies. One of the most striking findings of the survey was the prevalence of deep learning and context-aware recommendation techniques, which have long been considered better than more traditional methods. At last, offered detailed findings from the quantitative evaluation of publications that might help academics comprehend the present state and future prospects of RSs in e-learning that are based on deep learning.[28]

CONCLUSION AND FUTURE SCOPE

RL offers a powerful foundation for recommender systems by enabling continuous adaptation through iterative interaction modeling, contextual awareness, and long-term optimization of user engagement. This study systematically explored foundational RL algorithms, DRL architectures, and hybrid approaches, demonstrating their ability to enhance personalization, capture dynamic user preferences, and balance exploration-exploitation trade-offs effectively. Integrating RL with semantic enrichment, fairness constraints, and hybrid content–CF techniques has been shown to improve recommendation diversity, scalability, and resilience in dynamic and non-stationary environments. Computational complexity, cold-start scenarios, data sparsity, and restricted interpretability are some of the hurdles that still prevent widespread industrial usage, even with recent developments.

Future research should focus on designing computationally efficient RL algorithms suitable for large-scale recommendation systems while maintaining accuracy and responsiveness. Advancing explainable RL models will be critical to improving transparency, fostering user trust, and multi-agent RL approaches hold promise for bridging the gap between research innovations and practical deployment.

REFERENCES

1. Mosavi A, Faghan Y, Ghamisi P, Duan P, Ardabili SF, Salwana E, *et al.* Comprehensive review of deep reinforcement learning methods and applications in economics. Mathematics 2020;8:1640.
2. Kaminskas M, Bridge D. Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. ACM Trans Interact Intell Syst 2016;7:1-42.
3. Balasubramanian A. Personalized learning style detection and pathway optimization using hybrid machine learning approaches. Int J Sci Res Eng Manag 2025;9:1-7.
4. Afsar MM, Crump T, Far B. Reinforcement learning based recommender systems: A survey. ACM Comput Surv 2022;55:1-38.
5. Balasubramanian A. AI-Enabled demand response: A framework for smarter energy management. Int J Core Eng Manag 2018;5:96-110.
6. Bauer C, Zangerle E, Said A. Exploring the landscape of recommender systems evaluation: Practices and perspectives. ACM Trans Recomm Syst 2024;2:1-31.

7. Gao C, Lei W, He X, De Rijke M, Chua TS. Advances and challenges in conversational recommender systems: A survey. AI Open 2021;2:100-26.
8. Zhou S, Dai X, Chen H, Zhang W, Ren K, Tang R, *et al.* Interactive Recommender System via Knowledge Graph-enhanced Reinforcement Learning. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval; 2020. p. 179-88.
9. Chen X, Yao L, McAuley J, Zhou G, Wang X. Deep reinforcement learning in recommender systems: A survey and new perspectives. Knowl Based Syst 2023;264:110335.
10. Pandya S. Comparative analysis of large language models and traditional methods for sentiment analysis of tweets dataset. Int J Innov Sci Res Technol 2024;9:1647-57.
11. Lin Y, Liu Y, Lin F, Zou L, Wu P, Zeng W, *et al.* A survey on reinforcement learning for recommender systems. IEEE Trans Neural Networks Learn Syst 2024;35:13164-84.
12. Raza S, Rahman M, Kamawal S, Toroghi A, Raval A, Navah F, *et al.* A Comprehensive Review of Recommender Systems: Transitioning from Theory to Practice; 2025.
13. Gao C, Zheng Y, Li N, Li Y, Qin Y, Piao J. A survey of graph neural networks for recommender systems: Challenges, methods, and directions. ACM Trans Recomm Syst 2023;1:1-51.
14. Krishnamoorthi S, Shyam GK. Review of Deep Reinforcement Learning-Based Recommender Systems. In: 2022 Third International Conference on Smart Technologies in Computing, Electrical and Electronics (ICSTCEE); 2022. p. 1-12.
15. Patel D. AI-enhanced natural language processing for improving web page classification accuracy. ESP J Eng Technol Adv 2024;4:133-40.
16. Majumder RQ. Machine learning for predictive analytics : Trends and future directions. Int J Innov Sci Res Technol 2025;10:4.
17. Shahbazi Z, Jalali R, Shahbazi Z. Enhancing recommendation systems with real-time adaptive learning and multi-domain knowledge graphs. Big Data Cogn Comput 2025;9:124.
18. Rongala S, Pahune SA, Velu H, Mathur S. Leveraging Natural Language Processing and Machine Learning for Consumer Insights from Amazon Product Reviews. In: 2025 3rd International Conference on Smart Systems for Applications in Electrical Sciences (ICSSES); 2025. p. 1-6.
19. Chen X, Yao L, McAuley J, Zhou G, Wang X. A survey of deep reinforcement learning in recommender systems: A systematic review and future directions. J ACM 2021;37:2.
20. Zhang K, Cao Q, Sun F, Wu Y, Tao S, Shen H, Cheng X. Robust recommender system: A survey and future directions. ACM Comput Surv 2025;1:3.
21. Liu F, Tang R, Li X, Zhang W, Ye Y, Chen H, *et al.* Deep Reinforcement Learning Based Recommendation with Explicit User-Item Interactions Modeling [Preprint]; 2018.
22. Yan C, Xian J, Wan Y, Wang P. Modeling implicit feedback based on bandit learning for recommendation. Neurocomputing 2021;447:244-56.
23. Kalideen MR, Yağli C. Machine learning-based recommendation systems: Issues, challenges, and solutions. J Inf Commun Technol 2025;2:6-12.
24. Boka TF, Niu Z, Neupane RB. A survey of sequential recommendation systems: Techniques, evaluation, and future directions. Inf Syst 2024;125:102427.
25. Liu H, Cai K, Li P, Qian C, Zhao P, Wu X. REDRL: A review-enhanced deep reinforcement learning model for interactive recommendation. Expert Syst Appl 2023;213:118926.
26. Lin Y, Liu Y, Lin F, Zou L, Wu P, Zeng W, *et al.* A Survey on Reinforcement Learning for Recommender Systems [Preprint]; 2023. p. 1-21.
27. Wu S, Sun F, Zhang W, Xie X, Cui B. Graph neural networks in recommender systems : A survey. ACM Comput Surv 2022;55:97.
28. Salau L, Hamada M, Prasad R, Hassan M, Mahendran A, Watanobe Y. State-of-the-art survey on deep learning-based recommender systems for E-learning. Appl Sci 2022;12:11996.