A Survey on the Application of Reinforcement Learning in Recommendation Systems

Siddhanth Darshan Jain Gouder Nagpal *

Posted Date: 24 May 2025

doi: 10.20944/preprints202505.1892.v1

Keywords: Reinforcement Learning; Recommendation Systems; Deep Q-Networks; Policy Learning; Knowledge Graphs; Hierarchical Reinforcement Learning; Sequential Decision Making

 doi:10.20944/preprints202505.1892.v1

Article

A Survey on the Application of Reinforcement Learning in Recommendation Systems

Siddhanth Darshan Jain Gouder Nagpal

Portland State University; siddhan@pdx.edu

Abstract: From media streaming and e-commerce to education and healthcare, recommendation systems are now absolutely essential in many different fields. Conventional methods including content-based filtering and collaborative filtering sometimes miss the sequential, changing character of user preferences. By simulating recommendations as sequential decisions with long-term feedback, reinforcement learning (RL) offers a strong substitute. This survey presents a thorough investigation of RL-based recommendation systems together with important frameworks including hierarchical reinforcement learning, policy-guided reasoning, and Deep Q-Networks. We provide a disciplined taxonomy contrasting these approaches by design, flexibility, and application setting. We also look at ethical issues, pragmatic deployment problems, and evaluation difficulties in actual environments. By mapping the changing terrain of RL in recommendation and pointing up future directions, this work seeks to direct practitioners as well as researchers.

Keywords: reinforcement learning; recommendation systems; deep Q-networks; policy learning; knowledge graphs; hierarchical reinforcement learning; sequential decision making

---

1. Introduction

1.1. History and Motivation

With consumers overflowing with options across platforms, from e-commerce to online streaming and digital education, recommendation systems (RS) have become indispensable tools for content filtering and personalized decision support in the digital age. An RS seeks relevant objects for users by means of historical data, preferences, and behavioral signals. Early recommendation systems filtered items mostly using heuristic based approaches including simple content matching and popularity scores. Even in absence of explicit content features [3], letting systems learn patterns from user item interactions changed the field over time. On sites like Amazon and Netflix, where customizing the user experience directly resulted in more user involvement and income, these systems were especially crucial.

As the Web 2.0 paradigm evolved, RS sank more deeply into the fabric of daily digital interactions. From customized video queues on YouTube to tailored music playlists on Spotify and Twitter, recommendation systems changed the way consumers find and consume content. RS have also expanded into education, e-commerce, and healthcare domains [2,10]. RS reduces search friction, increases satisfaction by customizing content to user preferences, and provides businesses with a competitive edge, so reflecting their efficacy. These applications demonstrate how central RS is in guiding user paths across several domains.

Traditional RS have several natural constraints even if they are rather successful. Most people think that user preferences either change gradually or are fixed and depend mostly on stationary datasets. Sometimes they cannot adapt in real time to dynamic surroundings or reason over complicated relational systems. Depending on current activity, time of day, or even outside events, for example, a user's interest in instructional materials may change. Moreover, conventional models such as matrix factorization or nearest neighbor techniques do not naturally reflect the sequential character of user behavior, in which one interaction shapes next decisions [6]. These gaps become more clear as platforms give user retention and long term interaction top priority over transient clicks or ratings.

 doi:10.20944/preprints202505.1892.v1

1.2. Limitations of Traditional Models

Operating mostly under the supervised learning paradigm, traditional recommendation systems treat recommendation as a pointwise, pairwise, or listwise prediction task. Usually driven by historical user-item interaction data, these systems predict outcomes including preferences, clicks, or ratings. Among popular models are matrix factorization, nearest-neighbor algorithms, and more recently deep learning-based architectures including autoencoders and convolutional neural networks. Although these models have great accuracy in offline evaluations, they make static assumptions mostly that user preferences remain constant over time and that the objective is to maximize immediate predictive accuracy [6,9]. Consequently, conventional approaches cannot incorporate the long-term effects of recommendations or interactively change during deployment.

1.3. Why Reinforcement Learning?

The need of modeling dynamic, sequential interactions with consumers becomes more critical as recommendation systems develop beyond stationary personalization engines. Usually based on past data, traditional recommendation models generate one-shot predictions optimizing for instantaneous accuracy measures such as click-through rate or rating prediction. These approaches, meanwhile fail to consider how present recommendations might affect future user behavior, interests, or system confidence. By contrast, reinforcement learning (RL) is intrinsically suited to handle sequential decision-making, where the aim is to learn a policy that maximizes long-term cumulative reward via trial-and-error interactions with an environment [1,4]. This is well-suited with the way users interact in recommendation systems, where choices taken now can influence future preferences and loyalty of a user.

User comments are few and delayed in many practical applications. A user may click on a suggested item but only show long-term satisfaction by later actions including extended use, frequent visits, or downstream purchases. By means of its capacity to learn from delayed and sparse rewards, RL offers a principled framework to manage such situations. Suggesting a niche artist on a music recommendation system, for example, may first seem less than ideal, but if it helps the user explore and interact more deeply with the platform, it is eventually a good thing. RL lets recommendation agents find such policies maximizing for longer-term user satisfaction and platform utility [5,10]. Adopting RL in recommendation environments is much motivated by this ability for modeling temporal dependencies and delayed outcomes.

The capacity of RL to strike a balance between exploration and exploitation adds another important benefit. Traditional systems sometimes suffer from the so called filter bubble problem, whereby users are constantly shown similar content based on past preferences, so perhaps limiting discovery and long-term satisfaction. Strategic exploration of new content or categories by RL algorithms helps learning which objects may reveal latent user interests. This is particularly helpful for long-tailed material underrepresented in historical interactions or for cold-start situations. Epsilon-greedy, Thompson sampling, or upper confidence bound (UCB) strategies let RL-based recommenders cleverly control uncertainty and increase personalization over time [8,9]. This makes RL especially appropriate for adaptive systems that have to run in settings with either partial or changing information.

RL also lets the surroundings be an always learning ground. While models of supervised learning demand constant retraining, RL agents can change their policies online as fresh data arrives. Fast changing industries like e-commerce or streaming services, where user behavior, preferences, and available items change very quickly, absolutely demand this ability. By allowing reinforcement learning systems to adjust to such dynamics without depending just on batch updates, greater responsiveness and personalization [2,6] can result. Rich contextual signals such as user mood, device type, or session time can also be included into RL frameworks' state representations, so enabling more complex and situationally aware recommendations. This flexibility supports the increasing agreement among experts that reinforcement learning is not only an improvement but also a fundamental paradigm for the upcoming intelligent recommendation systems.

 doi:10.20944/preprints202505.1892.v1

1.4. Key Contributions of This Survey

This survey aims to go beyond summarizing existing work by offering the following contributions:

- We provide a structured taxonomy of reinforcement learning frameworks applied in recommendation systems, categorized by decision structure, adaptability, and application focus.
- We critically compare the strengths, weaknesses, and deployment contexts of value-based, policy-based, and hierarchical RL methods.
- We highlight key challenges in real-time personalization, fairness, and reward modeling, proposing emerging directions such as offline RL and hybrid policy learning.
- We contextualize reinforcement learning applications beyond e-commerce, with attention to healthcare, education, and high-stakes decision domains where interpretability and safety are paramount.

2. Background and Foundations

2.1. Fundamentals of Reinforcement Learning

A learning paradigm called reinforcement learning (RL) emphasizes how agents might learn to make decisions by interacting with an environment. The agent sees a state, acts, gets feedback in the form of a reward, then moves to another state. It develops a strategy or policy that guides its decisions on actions maximizing the overall reward it can acquire over time. Usually modeled using a Markov Decision Process (MDP), this interaction shows future states depending just on the current state and action, not on past history [3]. In environments where rewards are few or delayed, RL is especially effective since it allows one to learn from feedback instead of labeled data.

Starting with basic algorithms such as Q learning and SARSA, which learn value functions estimating the expected return of performing specific actions in particular states, the field has changed fundamentally. Policy gradient techniques particularly for continuous or high dimensional action environments became rather well known as RL developed. These approaches directly maximize the decision making policy by applying gradients of the expected return. More recently, deep reinforcement learning has merged RL with deep neural networks to manage challenging input spaces such as images, graphs, or sequences. Notable examples include actor critic approaches like A3C and PPO and Deep Q Networks (DQN), which have shown success in a broad spectrum of uses [1,8].

The flexibility of RL in modeling decision making over time attracts research in fields including recommendation systems for one of the main reasons. RL takes long term action consequences into account unlike more conventional models that generate isolated predictions. It is also meant to balance exploration, trying new activities to find better strategies, with exploitation, choosing the most well known activities so far. In user facing applications, where the system has to choose whether to suggest known content or introduce novelty, this balance is particularly crucial. Based on user interaction and feedback, RL provides a natural and strong means of always improving decision making in such dynamic surroundings [4,5].

2.2. Modeling Recommendations as Markov Decision Processes

Framing the problem as a Markov Decision Process (MDP) is among the most important conceptual change in contemporary recommendation research. This viewpoint lets the system replicate the sequential and interactive character of user involvement, in which every recommendation affects not only an instantaneous action such as a click or a rating but also the user's future preferences and behavior. The recommendation system is handled in an MDP formulation as an agent who observes the current state (e.g., user history, context), takes an action (e.g., recommends an item), and gets a reward (e.g., a click or watch time), before switching to a new state. The agent gradually learns to maximize long term cumulative rewards instead of only instantaneous feedback [1,4].

There are many advantages from this change of viewpoint. It first helps the system to reason regarding delayed feedback. A user might not react right away to a recommendation, for example, but it could affect future behavior including visiting the platform or investigating fresh content. Second,

 doi:10.20944/preprints202505.1892.v1

it promotes techniques that strike a mix between long term retention and temporary gratification, something conventional models are not meant to manage. Third, the MDP framework can include contextual cues including session length, device type, time of day, and user demographics, so providing a more complex picture of user behavior [5,8]. These features are particularly important on systems where user satisfaction develops over several sessions or actions.

Several more recent studies have shown how well this MDP based perspective performs. Policy learning techniques have been used, for instance, to investigate material in a more ordered manner; models such as PGPR use MDPs to negotiate knowledge graphs for goal directed, explainable recommendations [1,2]. By dynamically prioritizing more informative user interactions, real time adaptive models including IDEM DQN demonstrate how the MDP framework might guide learning in fast changing environments [8]. Modeling recommendations as MDPs has evolved into a basic first step toward creating really interactive and intelligent recommendation agents as the field keeps exploring this direction.

3. Reinforcement Learning Frameworks in Recommendation

3.1. Policy-Guided Path Reasoning over Knowledge Graphs

Knowledge graphs (KGs) offer relational, ordered information that guides users, objects, and auxiliary entities such as brands, categories, actors along interpretable paths. Including KGs into recommendation systems helps models to capture semantic and contextual signals often missed by cooperative or content based approaches, and enable them to reason over multi hop relationships. By allowing an agent to actively explore the graph, learning which reasoning paths are most helpful for producing high quality, explainable recommendations, reinforcement learning adds still another level of depth [1,2].

In this field, a representative approach is the Policy Guided Path Reasoning (PGPR) framework, which views recommendation as a path finder activity across a KG. Beginning at a user node, the RL agent learns to negotiate the graph via a series of relations to reach possible item nodes. Using reward signals reflecting both relevance and interpretability, the learnt policy directs this traversal. PGPR explicitly models the reasoning process, unlike embedding based techniques depending on latent similarity scores, so offering interpretable paths as justification for every recommendation [1]. By means of meaningful pattern discovery in the graph structure, this not only improves transparency but also enables the system to generalize better between various user item pairs.

In this context, RL's capacity to selectively investigate the combinatorially vast space of possible paths makes it especially potent. While still learning, RL agents can avoid noisy or irrelevant branches of the graph by using techniques including soft rewards and action pruning, so covering effective reasoning chains. Furthermore, policy networks learned on large scale knowledge graphs can adjust to changes in the underlying graph structure or user behavior, so strengthening and scaling these models for practical use. Especially in fields with sparse interactions and rich side information, recent research has shown that path based RL methods outperform conventional graph embedding approaches in both recommendation accuracy and explainability [2,4].

3.2. Hierarchical Reinforcement Learning for Structured User Goals

Single layer RL policies often find it difficult to reflect such hierarchical intent as recommendation tasks get more complicated, particularly in cases where user goals are multifarious or change with time. This restriction has led to hierarchical reinforcement learning (HRL), a paradigm whereby decision making is arranged at several layers. In an HRL based recommender, a high level policy might first decide on a coarse grained user objective such as selecting a genre, product category, or content theme while a low level policy then makes fine grained decisions such as choosing a specific item within that category [6]. This division of concerns enables the system to represent both abstract objectives and detailed actions, so enhancing interpretability and planning effectiveness.

 doi:10.20944/preprints202505.1892.v1

The hierarchical approach fits rather nicely how actual users interact with systems. In an e commerce environment, for instance, a user might begin with a broad intention like "buying electronics" then focus on a particular phone model. A flat policy would have to learn all such differences in a single environment, which becomes ineffective and prone to overfitting in large action environments. On the other hand, HRL arranges this learning process to mirror the layered character of decision making, so facilitating improved generalization between users and activities [9]. Furthermore, this configuration supports temporal abstraction, enabling lower level policies to control instantaneous interactions while high level policies run over longer time horizons, so providing a more natural fit for recommendation tasks spanning several sessions or behavioral phases.

From a learning standpoint, HRL lowers the decision space at every level so enhancing exploration. Low level policies refine the choice; high level policies direct investigation toward interesting areas of the item space. Since high level strategies can be used or refined independently of low level decisions, this breakdown also facilitates the transfer of policies across domains or user segments. Some models expand this concept by adding memory modules or attention based mechanisms to enable policies to alternate between strategies depending on user context or feedback history [5]. Particularly in dynamic and goal driven environments, empirical studies have found that HRL based recommenders not only increase long term user engagement but also produce more consistent and interpretable interaction patterns.

3.3. Adaptive Deep Q-Networks for Real-Time Personalization

User preferences in many real-world recommendation systems can change quickly depending on recent interactions, temporal context, or changes in the available content. Training offline and updated periodically, static models are not suited to handle such variations. Originally designed for control tasks in high-dimensional environments, Deep Q-Networks (DQNs) present a feasible framework for modeling user interaction as a real-time decision process. Using a neural network, a DQN approximates the action-value function, so allowing it to estimate the expected future reward of recommending an item given a user's current state [8]. By iterative updates, it learns to make ever more accurate decisions over time, balancing exploration of new possibilities with use of known preferences.

Standard DQNs in recommendations are mostly constrained by their presumption of a rather stable surroundings. This assumption is hardly true in dynamic platforms such as streaming services or online markets. To handle this, researchers have developed adaptive versions of DQN that can better react to environmental change. One such example is IDEM-DQN since it offers a dynamic experience sampling mechanism that gives transitions depending on environmental feedback and learning progress top importance. Unlike treating all past interactions equally, the model stresses significant events that offer better learning signals under current conditions [8]. This especially helps learning efficiency and stability when user behavior is erratic or feedback is delayed.

Adaptive DQN systems allow systems to personalize recommendations in real time without depending on perfect retraining, so enabling online updates. These systems fit applications including news feeds, e-learning platforms, and real-time product recommendations since they always more precisely reflect user interests. Moreover, their modular design lets one include in the state representation additional contextual signals including location, time of day, or device type. This provides a more whole picture of the user's present intent and supports quite exact personalizing approaches. Since they let one learn straight from user feedback and change in real time, DQN-based models are a great tool for offering timely and relevant recommendations in dynamic environments [4,5].

3.4. Comparative Summary of RL Frameworks

Deep Q-Networks (DQN) value-based approaches known for efficient online learning and quick decision-making are more suited for uses like news or video recommendations even though they struggle with delayed or sparse rewards. Policy-Guided Path Reasoning (PGR) investigates knowledge graphs using a policy-based approach and offers interpretable, ordered recommendations, although depending on well defined domain knowledge. Although Hierarchical Reinforcement Learning (HRL)

 doi:10.20944/preprints202505.1892.v1

adds considerable training complexity, it allows multi-level decision-making to replicate difficult user goals across sessions. Adaptive DQNs such IDEM-DQN dynamically change to fit changing surroundings and user behavior, so offering strong real-time personalizing at the expense of more tuning effort.

4. Methodology

4.1. Evaluation Frameworks and Metrics

Analyzing recommendation systems, especially those motivated by reinforcement learning, requires a sophisticated approach beyond traditional accuracy measurements. Long standing measures of a system's ability to forecast the next item a user might interact with include Precision, Recall, Mean Reciprocal Rank (MRR), and Normalized Discounted Cumulative Gain (NDCG). These measures are useful in stationary environments since they provide understanding of the relevance of particular recommendations; but, they do not adequately reflect the changing, sequential character of user interactions [6,9].

New evaluation priorities brought forward by reinforcement learning focus on measures such as cumulative reward, average return per session, or retention based signals since an RL agent maximizes long term results by interacting over time. These steps enable one to evaluate whether the agent is acquiring strategies that keep users interested not just now but over time. An RL based video recommender might, for instance, give content that gradually generates interest top priority instead of merely suggesting viral clips that grab short term attention but discourage consistent use [4,5]. Under such circumstances, success is defined by patterns of continuous interaction rather than by a single click.

Furthermore crucial is to assess the qualitative aspects of the user experience. If a highly accurate system keeps suggesting like-minded or too popular products, it can still fall short due to a filter bubble impact. Diversity, novelty, serendipity, and coverage are among the metrics that help guarantee the system searches the item space in meaningful ways and surfaces material users might not have come across otherwise. Fairness is also becoming a major factor since it guarantees that suggestions do not routinely underrepresent particular users or content kinds. These more general measures are particularly important in RL environments, where improper guidance of exploration policies may unintentionally induce bias [1,10].

4.2. Practical Implementation Considerations

Constructing and assessing recommenders based on reinforcement learning requires several sensible trade off. Designing a simulation or offline environment where the agent might safely investigate several policies without impacting actual users is a common first step. Many studies model user feedback using logged interaction histories from public datasets including MovieLens, Yelp, or Amazon reviews. Although handy, these offline configurations are intrinsically limited; real world feedback often is more noisy, context dependent, and delayed than what simulations can record [5,8]. To validate model behavior in real world settings, many researchers thus finally migrate to online A/B testing or batch learning from bandit feedback.

Another important factor is the learning architectural choice. When rewards can be precisely calculated, value based approaches such as Deep Q Networks (DQN) are helpful; yet, they sometimes depend on stable and well defined surroundings. More flexibility and better fit for complex or continuous action environments are provided by policy based approaches including actor critic algorithms. Some systems combine in hybrid configurations using the stability of value functions with the expressiveness of learned policies [4]. Stability and sample efficiency remain fundamental problems regardless of the approach, especially since most RL algorithms need thousands of interactions to converge, which may not be feasible in recommendation settings.

Ultimately, system performance can be much affected by technical choices on policy updates, reward modeling, and exploration exploitation balance. Rewards might come from explicit comments

 doi:10.20944/preprints202505.1892.v1

like clicks or ratings or from implicit signals like dwell time, scroll behavior, or perhaps return visits. While still learning from new interactions, exploration strategies must be precisely tuned to prevent overwhelming users with pointless material. One can achieve this by means of epsilon greedy, entropy regularization, or prioritized experience replay. As RL systems are used in high impact applications, careful design of their learning loops and the environments in which they are tested will remain fundamental to ensuring they are not only smart but also responsible and efficient [2,7].

5. Discussion

5.1. Core Insights and Practical Lessons

Growing evidence over the past few years shows that reinforcement learning can fundamentally alter the design and optimization of recommendation systems. One of the most obvious realizations is that, instead of independent predictions, modeling recommendations as sequential decisions opens the path to optimizing for long term engagement, not only one time clicks or ratings. This change lets systems take user satisfaction into account over sessions and modify their plans depending on changing preferences [1,4]. On both offline benchmarks and online platforms, many RL based systems including those developed on Deep Q Networks, policy gradients, and multi agent architectures have shown good performance.

Still another key lesson is the flexibility RL offers in combining structured knowledge with behavioral learning. Techniques including PGPR show how reinforcement learning can be used for reasoning, navigating knowledge graphs and producing interpretable recommendations [1,2] in addition to making predictions. Hierarchical and modular RL systems show how abstract user goals might be obtained alongside low level item selection, so enabling systems to better match the multi intent character of actual user sessions [6]. Actually, these models also enhance personalization since the agent gains knowledge from user specific trajectories instead of depending just on aggregated worldwide data.

Although RL algorithms are complex, practical implementations show that many systems gain even from rather basic RL policies when tuned correctly. Often outperforming more complex but brittle baselines are lightweight models with well designed reward functions and exploration strategies. Furthermore, RL based recommenders are especially suited for contemporary digital environments because of their capacity to personalize in real time, adjust to feedback on the fly, and even apply acquired policies across multiple domains. These lessons imply that even if RL might not be a universal solution, it is a convincing direction for recommendation systems that must learn constantly and behave strategically [5,9].

5.2. Ongoing Challenges in RL-based Recommendation

Although RL has great potential for recommendation, several factors still restrict its acceptance and dependability in use. One of the most enduring problems is delayed and meager rewards. Many systems allow for either rare or only observable much later useful feedback from users, including clicks, purchases, or return visits. This makes it challenging for RL agents to reasonably estimate value or link results to particular recommendations. Designing reward signals that capture significant long term goals without requiring dense supervision remains an open research topic [7,10].

Another issue is scalability, especially for massive systems including millions of users and objects. Standard RL methods can find it challenging to converge or generalize effectively depending on the size of the state and action space. In high dimensional settings where user states combine behavioral history, context, and temporal dynamics, this difficulty is especially acute. Although they often require major engineering effort and tuning [6,8], techniques including experience replay, hierarchical policies, and approximative value functions help mitigate this.

Finally, before RL can be generally trusted in recommendation systems, interpretability and stability must be solved. Many RL models function as black boxes, thus it is difficult to justify why a given item was advised. In high stakes fields like education or healthcare, this lack of openness can

 doi:10.20944/preprints202505.1892.v1

hinder acceptance. Furthermore sensitive to hyperparameters and prone to instability during training, especially in settings where user behavior is quite nonstationary, are RL systems. Dealing with these issues will call for fresh approaches for explainable policy learning, safe exploration, and more strong evaluation systems [2,4]. Although these difficulties are not insurmountable, they draw attention to the need of constant research and ethical design principles.

6. Future Directions and Open Challenges

Along with a set of challenging but crucial open questions, several exciting directions are starting to show up as reinforcement learning develops inside the recommendation terrain. Improved matching of reinforcement learning goals with human-centered objectives is one important area of future development. While many RL agents are still instructed to maximize technical indicators such cumulative reward or engagement time, these do not always reflect what users really value: confidence, satisfaction, or meaningful discovery. Still quite challenging is designing reward functions reflecting user intent more holistically, maybe using implicit behavioral cues or preference modeling.

Making RL-based systems more data-efficient and strong in practical environments is another fascinating direction. Although deep RL algorithms have shown remarkable performance in controlled environments, they sometimes require thousands of interactions to converge something that is hardly useful in live systems where user experience counts. Using methods including offline policy learning, meta-learning, or model-based RL could help to lower the data load and raise sample efficiency. To speed learning and stabilize training dynamics, hybrid architectures combining RL with supervised learning, knowledge graphs, or pre-trained user embeddings also attract increasing interest.

Whether it's in learning platforms, healthcare advice, or even news exposure, as recommendation systems get more entwined into daily decision-making, ethical questions of RL also become increasingly important. For example, exploration techniques must be handled carefully to prevent surfacing biassed or damaging material in the name of learning. Especially in high-stakes fields, more openness, interpretability, and user control are also much needed. Fostering trust and long-term involvement will depend on building systems that let users grasp, calibrate, or even affect the recommendation process.

Ultimately, many unresolved issues surround how to scale RL to vast, varied user bases without compromising fairness or personalization. Policies should be able to provide recommendations at an individual level yet also generalize across users. As content catalogs, user behavior, and feedback loops always change, how can systems stay stable? Dealing with these problems calls for multidisciplinary approaches combining developments in system design, human-computer interaction, and machine learning. Reinforcement learning will help to shape the next generation of recommendation systems as we go forward; but, realizing vision will need careful innovation based on both technical rigor and user-centered thinking.

7. Conclusions

7.1. Final Summary

This survey aims to investigate the evolving contribution of reinforcement learning to the design and implementation of modern recommendation systems. We began by assuming the limits of traditional approaches, which often see recommendation as a stationary prediction task and struggle to account for sequential decision making, long term user engagement, or dynamic environments. Reinforcement learning offers a strong alternative by modeling the recommendation process as an interactive loop, one in which every action shapes future user behavior and system outcomes.

We demonstrate among other RL based models how Deep Q Networks, policy gradient methods, hierarchical RL, and knowledge graph traversal let systems learn from experience, adapt over time, and maximize for more meaningful engagement signals. Beyond business venues, the efficient implementation of RL has attracted increasing momentum in fields including emergency response, education, and healthcare. These developments show how RL is a strategic transformation in how systems learn to personalize, reason, and grow, not only a performance booster.

 doi:10.20944/preprints202505.1892.v1

7.2. The Road Ahead for RL in Recommendation Systems

As reinforcement learning develops, its inclusion into recommendation systems offers both amazing possibilities and urgent problems. Future study has to address pragmatic constraints including low feedback, high sample complexity, and scalable architecture needs. Broader issues of fairness, openness, and ethical inquiry especially in high-impact applications where recommendations affect learning outcomes, healthcare decisions, or resource allocation have great relevance.

Strong momentum toward more data-efficient and responsible RL systems including offline learning, model-based approaches, and human-in- the-loop feedback mechanisms exists. Furthermore, we expect more cross-pollination among reinforcement learning and other disciplines including causal inference, preference modeling, and HCI. Recommendation systems' future is ultimately in creating agents that are not only accurate but also trustworthy, flexible, and in line with user needs. Thoughtfully applied reinforcement learning is positioned to enable realization of that vision.

References

1. Yikun Xian, Zuohui Fu, S. Muthukrishnan, Gerard de Melo, and Yongfeng Zhang. Reinforcement knowledge graph reasoning for explainable recommendation. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2019.
2. Xiting Wang, Kunpeng Liu, Dongjie Wang, Le Wu, Yanjie Fu, and Xing Xie. Multi-level recommendation reasoning over knowledge graphs with reinforcement learning. *Proceedings of the ACM Web Conference*, 2022.
3. Pengyang Wang, Kunpeng Liu, Lu Jiang, Xiaolin Li, and Yanjie Fu. Incremental mobile user profiling: Reinforcement learning with spatial knowledge graph for modeling event streams. *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2020.
4. Dongjie Wang, Pengyang Wang, Kunpeng Liu, Yuanchun Zhou, Charles Hughes, and Yanjie Fu. Reinforced imitative graph representation learning for mobile user profiling: An adversarial training perspective. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.
5. Dongjie Wang, Pengyang Wang, Yanjie Fu, Kunpeng Liu, Hui Xiong, and Charles Hughes. Reinforced imitative graph learning for mobile user profiling. *IEEE Transactions on Knowledge and Data Engineering*, 2023.
6. Shipeng Guo, Kunpeng Liu, Pengfei Wang, Weiwei Dai, Yi Du, Yuanchun Zhou, and Wenjuan Cui. Rdkg: A reinforcement learning framework for disease diagnosis on knowledge graph. *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2023.
7. Lu Jiang, Kunpeng Liu, Dongjie Wang, and Pengyang Wang. Reinforced explainable knowledge concept recommendation in moocs. *ACM Transactions on Intelligent Systems and Technology*, 2023.
8. Yanan Xiao, Lu Jiang, Kunpeng Liu, Yuanbo Xu, Pengyang Wang, and Minghao Yin. Hierarchical reinforcement learning for point of interest recommendation. *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, 2024.
9. Xinhao Zhang, Jinghan Zhang, Wujun Si, and Kunpeng Liu. Dynamic weight adjusting deep q-networks for real-time environmental adaptation. *arXiv preprint arXiv:2411.02559*, 2024.
10. Kunpeng Liu, Xiaolin Li, Cliff C. Zou, Haibo Huang, and Yanjie Fu. Ambulance dispatch via deep reinforcement learning. *Proceedings of the 28th International Conference on Advances in Geographic Information Systems (SIGSPATIAL)*, 2020.