# Core Technologies in Recommender Systems: Investigating and Analyzing Standard Implementations

Yifan Cai

Shanghai Tech University, Shanghai, 201210, China

caiyf@shanghaitech.edu.cn

**Abstract.** This paper places special emphasis on the evolution of recommendation algorithms in the context of big data and machine learning, underscoring significant advancements in deep learning and natural language processing that have markedly improved the precision and personalization of recommendations. We explore case studies in various sectors, including e-commerce, streaming services, and social media, to illustrate the adaptation of these technologies to distinct industry requirements. A critical component of our analysis is the examination of the impact of user data privacy regulations on the design and functionality of these systems. Additionally, the paper addresses the challenges and prospective directions in the field, with a particular focus on ethical considerations, the mitigation of bias, and the incorporation of artificial intelligence for dynamic, context-aware recommendation systems. Our research methodology amalgamates an extensive literature review with empirical data analysis, offering an in-depth understanding of the current state of recommendation system technologies. The findings of this study are intended to contribute to the field by presenting a comprehensive view of the technological, ethical, and practical dimensions of recommendation systems in the contemporary digital landscape.

**Keywords:** Recommendation Systems; Collaborative Filtering; Content-based Filtering; Machine Learning in Recommendations.

## 1. Introduction

In the contemporary era, individuals are immersed in a data-rich environment of unprecedented scale. Big data offers a wealth of user information and intricate product details, enabling a comprehensive depiction of diverse entities. However, this abundance presents a challenge: the efficient utilization of vast, sparse, and variably qualitative data, which often results in information overload. Recommendation Systems (RS), a sophisticated form of information filtering technology, tackle this challenge by analyzing the interplay between user profiles and item characteristics. These systems are designed to tailor content to individual preferences, effectively sifting through superfluous information, and have emerged as a pivotal technology across various application domains [1]. The conceptual foundations of recommendation systems trace back to early explorations in information retrieval and filtering. The GroupLens system, introduced by Resnick in 1994, pioneered the application of collaborative filtering (CF) for recommendation tasks. In 1997, Resnick and colleagues provided a structured definition of recommendation systems, marking a significant milestone in the field's evolution [2]. Post-2000, major e-commerce platforms like Amazon and Netflix began incorporating these systems extensively, enhancing user experience and sales through personalized recommendations [3]. The advent of social media in 2010 catalyzed the adoption of users' social network information in recommendation algorithms, propelling the prominence of content-based systems [4-7]. Recent advancements have seen the integration of reinforcement learning into recommendation systems. This approach enables real-time user interactions to refine recommendation strategies, effectively addressing challenges such as the cold start problem, interpretability, and dynamic updating [8].

# 2. Overview of the Recommendation System

## 2.1. Basic concepts of the recommendation system

The core challenges addressed by Recommendation Systems encompass: mitigating the user's Long Tail Effect by catering to personalized, diverse, and niche demands of each user; and alleviating information overload through the extraction of pertinent information from extensive datasets. Presently, RS have progressively found broad and efficacious applications across various domains including shopping, news, education, and advertising, as detailed in Table 1.

**Table 1.** Specific Recommendation System Methods in Various Fields.

| Recommendation field | Classical system |
|---|---|
| Advertisement | SARSIS [9], Recogym [10] |
| News | DRN [11], MV-DNN [12] |
| Shopping | DeepPage [13], DeepCoNN [14] |
| Music and Movie | CKE [15], CDL [16] |

## 2.2. Technical Overview of Recommendation System

From a technological standpoint, the principal methodologies in recommendation systems are generally categorized into three distinct types: collaborative filtering-based recommendations, content-based recommendations, and hybrid recommendation approaches [17, 18].

### 2.2.1 Content-Based Recommendation

The algorithm commences by constructing vectors representing user preferences and item attributes, subsequently calculating their similarity to recommend items that exhibit the highest congruence with the user profile. The process is systematically outlined in Fig 1. Despite their high scalability and low complexity, content-based recommendation algorithms confront inherent limitations:

Limited Diversity in Recommendations: Predominantly reliant on the target user's historical data, content-based systems tend to offer recommendations with a relatively narrow scope. This reliance predisposes the system to suggest items akin to those previously interacted with by the user, potentially curtailing the diversity of recommendations presented.

Challenges with Product Feature Information: The management of voluminous and intricate product feature information poses a substantial challenge. An accurate content-based feature extraction often requires meticulous and diverse labels (tags). Content-based algorithms may falter when confronted with extensive and nuanced product attributes, hindering their ability to efficiently process and leverage such data.

Neglecting User Interest Evolution: A notable oversight of content-based recommendation algorithms is their tendency to disregard the dynamic nature of user interests. These systems often pay low attention to the temporal evolution of user preferences, and the large amount of historical data makes it difficult for the system to flexibly capture the latest user preferences, potentially leading to outdated or irrelevant recommendations.

High requirements for labels: The quality of the user’s and item’s label will greatly affect the recommended results. How to label the products requires strong expertise, especially for new products where algorithm can not often updated in time.

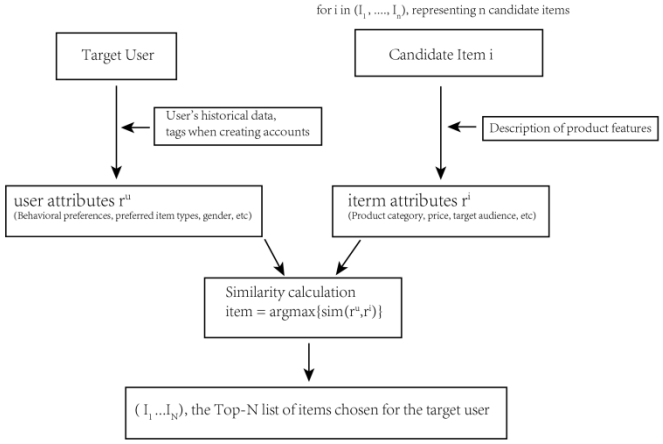


**Fig. 1** Structure diagram (Photo/Picture credit: Original).

### 2.2.2 Collaborative Filtering Techniques

The collaborative filtering-based technique capitalizes on the behavioral data between users, and correlation data between items. Within this category, item-based collaborative filtering algorithms focus on identifying items akin to the target item base on historical data. Subsequently, recommendations are made to users based on the similarity rankings of these items. This method typically resonates well with users' inherent preferences.

The user-based collaborative filtering algorithm, on the other hand, involves a series of distinct steps. It mainly focuses on finding the similar users with the target user and uses the preference from those users to create recommendation list. The process is systematically outlined in Fig 2, illustrating the sequential approach employed in this algorithm to generate user-specific recommendations. This methodology not only considers the ratings but also incorporates the relational dynamics among users to refine its suggestions.

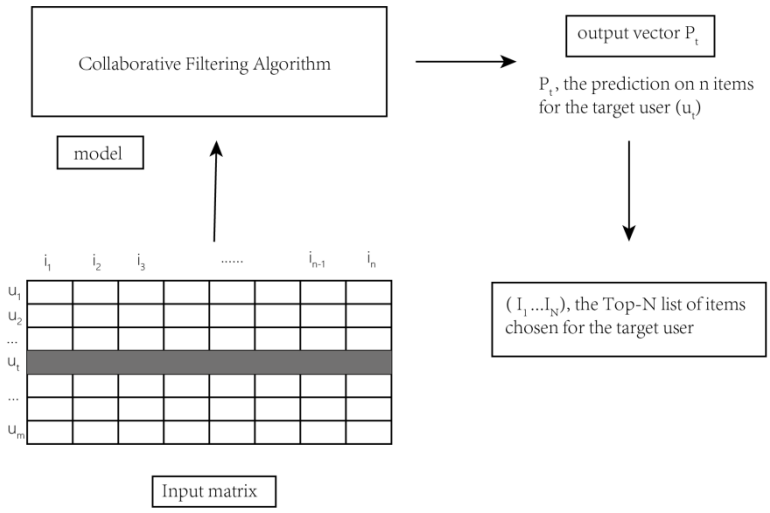


**Fig. 2** Structure diagram (Photo/Picture credit: Original).

The Input matrix (user rating matrix), where $U_i$ (i = 1, ..., m) represents m users and $i_n$ denotes the list of n items, $U_t$ is the target user who the algorithm will provide the recommendation. Based on similar users' preference, the favorite levels of the n-items are predicted. However, traditional collaborative filtering faces several challenges that need consideration. This method relies on the collection of historical user behavior data, encompassing actions such as user ratings, clicks, purchases, and more. It necessitates a sufficiently large user base and a diverse range of product categories. Yet, during the application process, several challenges are encountered, including but not limited to: Cold Start Problem, Sparse User Feedback Matrix, High-Dimensional Data.

### 2.2.3 Deep Learning in Recommendation Systems:

Deep learning, as a pioneering technology, facilitates automatic learning and intricate feature extraction from expansive datasets. These models leverage multiple layers of nonlinear transformations to autonomously discern abstract feature representations from input data, enabling them to model complex patterns and relationships with a high degree of sophistication. Prominent in this realm are deep recommendation models such as Wide & Deep, CDL, and AutoRec, among others.

In these models, the input layer integrates a variety of user attributes (e.g. age, gender, occupation), user feedback (e.g. clicks, ratings, browsing duration), and item attributes (e.g. item type, applicable crowd). The model layer, comprising a series of stacked deep learning models employing either linear or nonlinear transformations, is adept at learning latent representations of users and items. The output layer engages in similarity computation to retrieve a relevant subset of items from the item pool, subsequently employing a ranking algorithm to generate personalized recommendations [19]. The basic framework is shown in fig 3.

While these models have shown exceptional performance in diverse recommendation scenarios, deep learning has its limitations:

Cold Start Problem for new users and items: Deep learning's dependence on extensive historical data presents challenges in making accurate recommendations for new users or items with sparse historical data.

Lack of Interpretability: The end-to-end learning structure of deep learning models often results in a lack of interpretability. This opacity in the recommendation rationale can impede model refinement and user trust.

Dynamic User Preferences: In rapidly evolving environments, such as advertising and news recommendations where timeliness is paramount, the pre-defined processing patterns of deep learning models may fall short in accommodating evolving user preferences and real-time environment changes [20].

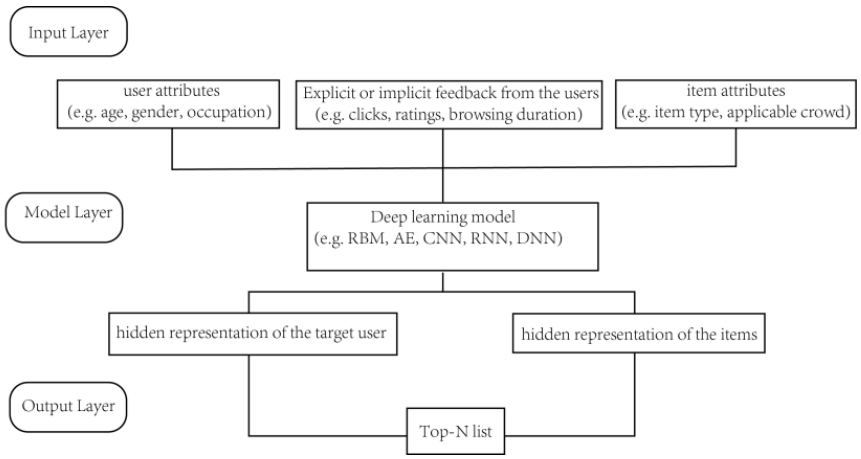


**Fig. 3** Framework of deep learning-based recommendation system (Photo/Picture credit: Original).

### 2.2.4 Application of Reinforcement Learning

As an innovative approach within the realm of Interactive Recommendation (IR), reinforcement learning-based recommendation systems adeptly uncover users' personal inclinations and the dynamic context of real-time interactions. These models, by continuously refining their recommendation strategies through user engagement, partially mitigate the cold start problem. This dynamic approach, which evolves user profiles during interactions, is more congruent with the realities of recommendation scenarios, as compared to traditional static methods [21]. Furthermore, the integration of exploration mechanisms within these systems facilitates users in discovering new interests, maintaining a positive receptivity towards novel products. This aspect is particularly pivotal in advertising (shopping) recommendations, where use’s taste might be stable, while the true needs fluctuates considerably over time.

The fusion of reinforcement learning with online and contextual recommendation methods has exhibited superior performance in making recommendations [22]. These models typically focus on maximizing cumulative rewards, aligning with long-term expectations and thereby fostering sustained user engagement with various platforms, such as news and music applications. Through extended interactions, these systems can more accurately portray user traits and establish nuanced product affiliations. For example, one might hardly shift from one music App to another after using for a long time, because the App can always not only recommend some of the music styles one used to like, but also reasonably recommend new songs that one have never touched, but meet one's taste, and this is very difficult to implement in new APP.

## 3. Analysis of Reinforcement Learning Techniques

### 3.1. Definition and Principles of Reinforcement Learning

Reinforcement Learning is intrinsically predicated on the paradigm of experiential learning, namely learning through trial and error. This process involves an agent operating within an environment (E). Herein, the agent, contingent upon its current state (S), undertakes actions (A) aimed at the maximization of rewards (R). Central to this approach is the iterative training of the agent to develop and refine policies (P), thus making RL a goal-directed learning methodology, anchored in interactive experiences. The basic framework is shown in fig 4.

The distinguishing characteristics of reinforcement learning encompass:

The real-time acquisition and assimilation of dynamic user preferences, facilitating adaptive response mechanisms.

The implementation of exploration strategies, specifically designed to mitigate the incidence of monotonous recommendations.

A pronounced focus on the enhancement of users' long-term satisfaction, ensuring that recommendations evolve in alignment with changing user behaviors and preferences.

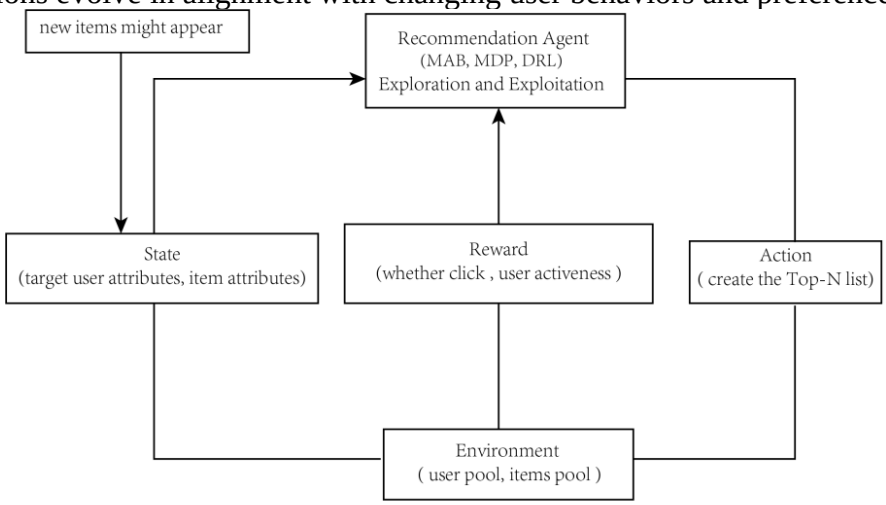


**Fig. 4** Framework of reinforcement learning-based recommendation system (Photo/Picture credit: Original).

### 3.2. Key Component of Reinforcement Learning: Reward Function

Its definition critically influences the effectiveness of algorithms, serving as an indicator of user satisfaction with recommended results. The ultimate goal is to maximize user satisfaction, which is maximizing rewards. In the realm of recommendation systems, addressing the challenge of obtaining a comprehensive reward metric is crucial due to the multi-channel nature of user evaluations and the specific requirements of recommendation systems (such as diversity, exploratory nature, and real-time considerations).

### 3.2.1 Click-Through Rate

Click-Through Rate is widely employed as an evaluation standard in recommendation systems [23]. CTR intuitively measures user interest in specific products, allowing for the scoring of various products. AI recorded user's click histories and recommends similar advertisements based on this information [24]. To refine the function, indicator functions such as clicks or non-clicks, can be transformed into multi-score ratings. Lu assigned scores of 2, 1, and 0 for user behaviors of forwarding or like, acceptance, and ignoring, respectively [25]. For platforms using likes and dislikes, a quantification of 1, 0, -1 for like, ignore, and dislike can be applied.

Some studies consider whether commenting can be regarded as an indicator of users' attention to advertisements and incorporated into CTR statistics. However, due to the user's careless error evaluation, or benefit-oriented malicious evaluation and forwarding, these metrics may not accurately reflect user preferences. Therefore, data reliability verification is typically necessary before model implementation.

### 3.2.2 Latent Preferences

In contrast to the direct evaluation of CTR, latent evaluations often stem from users' comments. Utilizing text analysis techniques, user sentiments towards products can be derived from comments. Models like Varlamis leverage user comments on review sites, employing techniques like latent semantic analysis and hierarchical clustering to discern user preferences [26]. Zheng proposed DeepCoNN for jointly learning project properties and user behavior from the comment text. Reviews were classified as positive and negative for inclusion in the reward criteria

### 3.2.3 Negative Feedback Data

In product and advertisement recommendations, negative feedback is typically more abundant than positive feedback. Therefore, analyzing negative feedback separately can help the system effectively avoid recommending products explicitly disliked by users. Zhao introduced the DEERS framework, a deep reinforcement learning framework that segregates positive and negative feedback as independent inputs, updating corresponding states. Experimental results on JD's dataset demonstrate its effectiveness [27].

### 3.2.4 Long-Term User Satisfaction

Zheng et al. proposed the DRN model, taking into account that the frequency of user interactions with the system serves as an indicator of user engagement, which in turn reflects the degree of user preferences for recommended content [11]. Consequently, they employed a DQN network to capture dynamic variations and utilized user activity as an additional metric. Zou et al. focused on the duration of user browsing and activity [28]. They innovatively introduced a Q-network named FeedRec to optimize the delayed feedback matrix for users, incorporating a hierarchical LSTM network. The devised reward function enables the model to simultaneously optimize immediate and delayed feedback. Rewards are represented through three matrices with weighted distributions: the click matrix (click metric), the browsing depth matrix (depth metric), and the access interval matrix (visit interval metric). The first matrix provides immediate feedback, while the latter two deliver delayed feedback.

### 3.2.5 Coverage Issue

Coverage metrics refer to the proportion of recommended items covering all items. Low coverage implies a limited choice of items for users [29].

Recommendation coverage represents the proportion of items recommended by the system to users among all items. If a recommendation algorithm consistently suggests items that users have already viewed, its coverage is often low, typically resulting in low diversity and novelty in recommendations. The main algorithms are $COV_P$ and $COV_R$:

Prediction coverage: $COV_P = \frac{N_d}{N}$.

where Nd represents the number of items the system can predict ratings for, and N is the total number of items.

Recommendation coverage: $COV_R(L) = \frac{N_d(L)}{N}$.

where L represents the recommendation list (Top-N list), $N_d(L)$ represents the number of distinct items that appeared in recommendation lists.

### 3.3. Main Algorithms of Reinforcement Learning

In the Multi-Armed Bandit paradigm, the environment's response to an agent's action is manifested through a reward system, which in turn informs and refines the agent's knowledge base. Central to the MAB framework is the duality of exploration and exploitation; it involves the pursuit of novel insights alongside the utilization of existing knowledge. MAB excels in recommending options that align with users' known preferences (exploitation) while simultaneously discovering potential new interests (exploration), thus alleviating the issue of repetitive suggestions and adding an element of unpredictability for the user.

Key traditional algorithms in the MAB framework include the Epsilon-Greedy algorithm, Thompson Sampling algorithm, Upper Confidence Bound algorithm, and the Contextual Multi-Armed Bandit (CMAB). Within this context, the agent operates under a specific policy π, basing its decisions on a probabilistic model of state transitions to select actions. The environment, in response, assigns a reward r based on these actions. Fundamental to this domain are algorithms like Q-learning and Sarsa, which are cornerstones of reinforcement learning.

Additionally, the strategy gradient-based algorithm merits attention for its direct method of inputting the state and outputting the probability of an action. This method assesses the relative effectiveness of an action through its associated reward. However, the updating of its parameters requires a complete cycle, highlighting the delicate interplay between learning efficiency and algorithmic complexity inherent in reinforcement learning.

### 3.4. Main Algorithms of Reinforcement Learning

In general, reinforcement learning techniques can be mainly divided into the following three aspects: Multi-Armed Band, MarkovDecisionProcess, Deep Reinforcement Learning.

#### 3.4.1 MAB

The Multi-Armed Bandit is a machine learning framework. In this system, an agent must make a choice from k actions (arms) during each round, aiming to maximize its cumulative reward over the long term. Every round involves the agent receiving information about the current state (context). Subsequently, an action is chosen based on this information and past experience from prior rounds. The environment then issues a reward tied to the selected action, prompting the agent to update its information. Essentially, the Multi-Armed Bandit presents a challenge centered around the delicate balance of exploration and exploitation.

MAB can not only recommend the product which is similar to user’ s previous interests based on exploitation but also discover the user’ s potential interests based on exploration which avoids repeated recommendation and bring wonderment to users.

Traditional MAB algorithms are Epsilon-Greedy algorithm, Thompson sampling algorithm, upper confidence bound algorithm and contextual Multi-armed Bandit (CMAB).

#### 3.4.2 Markov decision process

A Markov decision process is a discrete-time stochastic control process with five elements: state (s), action (a), reward (r), State transition probability matrix ($P^{\pi}_{st,st+1}$), discount factor r. The reinforcement learning based on MDP view the recommendation system as a Markov chain process which means the probability of each event depends only on the state attained in the previous event. The agent takes the policy π and bases on the state transition probability function to choose

an action. The environment will give a reward r based on the action. Basic algorithms are Q-learning and Sarasa.

### 3.4.3 Deep reinforcement learning

Deep reinforcement learning (DRL) combines the reinforcement learning and deep learning and can be divided into value based DRL and policy gradient based DRL.

The recommendation system based on the value function DRL uses deep neural networks to approximate the Q-value function (which is from Q-learning algorithm), with the optimization goal of maximizing the total reward. The neural network parameters are continuously updated through gradient descent to find the optimal strategy.

The strategy gradient based algorithm inputs the state and directly outputs the probability of an action. During training, the relative quality of an action is judged by the reward value, but its parameters must go through a complete turn before they can be updated.

## 4. Application of Reinforcement Learning

### 4.1. Application of Reinforcement Learning in Product Recommendation

Multi-Armed Bandit recommendation algorithms have been widely employed in the field of product recommendations, effectively addressing dynamic recommendation challenges in online scenarios. However, most of this research primarily focuses on improving recommendation accuracy, sacrificing the diversity and novelty of recommended results. This type of recommendation system tends to select products from an increasingly narrow range, such as popular items, leading to the issue of overspecialization, making it difficult to provide users with a fresh experience [30, 31].

In the study presented in reference, a Multi-Objective Interactive Recommendation System based on MAB (MOMAB) is proposed, considering accuracy, diversity, and novelty as simultaneous optimization objectives [32]. Initially, a user-based collaborative filtering algorithm is applied for Top-N filtering based on historical records, and the top 100 ranked products enter interactive reinforcement learning. Suppose we have a Top-N recommendation list P and A= $\{A_1, ..., A_N\}$ representing N MABs. The candidate recommended products is presented as a vector $I_m = \{i_1, i_2, \ldots, i_m\}$. An expected reward vector for each item $i_j$, $S_j = \{S_1^j, S_2^j, S_2^j\}$ for the three optimization objectives. Additionally, there is a weight vector W= $\{W_1, W_2, W_3\}$ representing objective weights. Inspired by the digital filtering process of the Kalman filter, a weight updating strategy is proposed. Using user feedback results from the t-th recommendation and the expected reward of the products$S_{m*p}$, the weight w for the (t + 1) th recommendation is updated, aiming to minimize the weighted regret of recommended products. The advantage of this method is that as the interaction frequency between users and the system increases, the recommended results increasingly meet user needs. Simultaneously, the algorithm accurately captures outdated products and changes in user interests, updating recommendation categories in real-time. For new users, the system prioritizes prediction accuracy to build trust with users, while for existing users, the system emphasizes diversity and novelty to broaden their perspective, bringing freshness and excitement, thereby enhancing user satisfaction and loyalty. The algorithmic process is roughly as follows:

This experiment will be conducted on the Hetrec2011-LastFM-2k dataset, and the experimental results significantly outperform the Ranked Bandit model (RB) and the Upper Confidence Bound Bandit model. As shown in Fig 5.

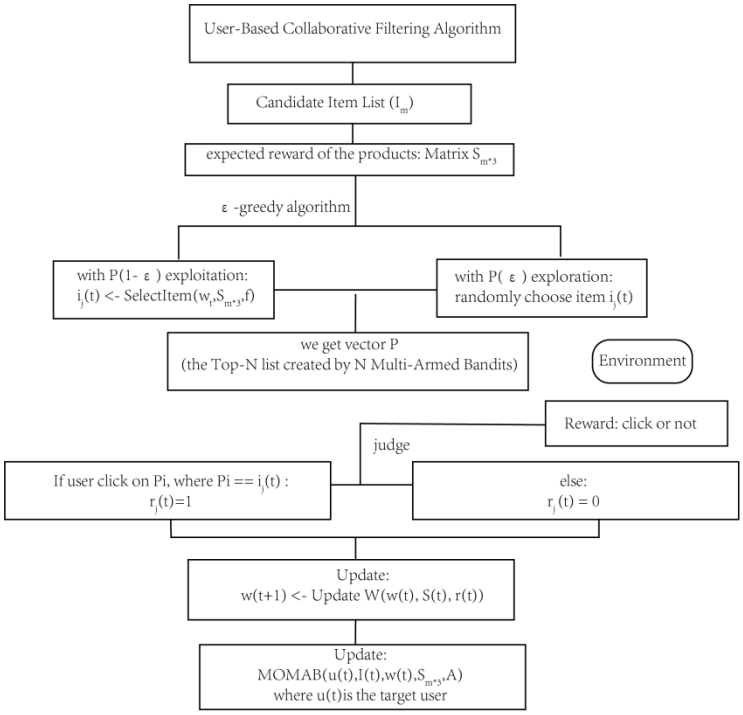


**Fig. 5** Framework of MOMAB (Photo/Picture credit: Original).

### 4.2. Application of Reinforcement Learning in the Field of Online Learning

In contrast to traditional recommendation systems such as advertisement, commodity, and news recommendations, online learning path recommendations require a comprehensive consideration of the differences in the learning abilities, knowledge backgrounds, and learning goals of target learners. It aims to tailor a learning path that adheres to educational principles and achieves the learner's specified learning goals. This process involves simultaneously considering the difficulty and progressive depth of materials, avoiding redundant recommendations of the same knowledge, and taking into account the student's expected learning cycle [33]. In reinforcement learning, the learner can be considered as the agent, learning resources as the environment, the learner's selection of learning resources as the agent's action, and the learning effect obtained by the learner after studying relevant knowledge as the reward from the environment to the agent's action. Recommending the optimal learning path for the target learner is the process by which the intelligent agent seeks a sequence that maximizes rewards in different environments.

The study presented in reference addresses the adaptive iterative selection of learning materials in the presence of imprecise information about learners and learning materials [34]. This problem is formulated within the framework of Markov Decision Processes, including a measurement model for evaluating student skill profiles, a learning model representing the effectiveness of learning materials, and a recommendation strategy mapping current available information to a given set of learning materials. The "what to learn" question depends on the content mastered so far, and the article adopts a cognitive diagnosis model as the measurement model to quantify the current learning status of students [35]. The effectiveness of each learning material is modeled through a Markov model, assuming that knowledge learned in the short term does not need to be repeated. The article employs a reinforcement learning approach based on Q-learning, balancing between possible optimal recommendation results and exploring new learning paths. Additionally, considering that the learning model may depend not only on skill profiles but also on other individual-specific factors such as knowledge acquisition ability, learning time constraints, and learning preferences (reading, solving problems, watching videos), these can be addressed by adding covariates (such as gender, age,

learning behavior) to parameterize the Q-learning function. Flow chart for basic recommendation system for personalized learning is as shown in Fig 6.

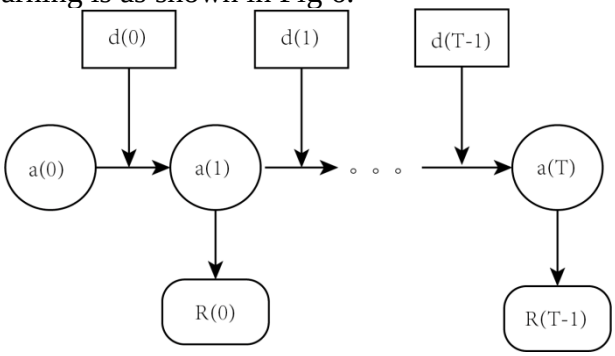


**Fig. 6** Framework of MOMAB (Photo/Picture credit: Original).

Consider a student with K skills to learn over T time periods (T time units). Let a(t) = [$a_1$(t), ..., $a_k$(t)] represent the proficiency levels of the student in each of the K skills at the beginning of learning stage t. At each time t, we recommend a set of learning materials d(t) from the learning material library, resulting in a transition from proficiency level a(t) to a(t+1). The optimization objective of the algorithm is to maximize the total expected number of skills mastered throughout the entire learning process. The immediate reward in epoch t is denoted as R(t), written as R(t)=$\sum_{k=1}^{K}[a_k(t+1)-a_k(t)]$.

### 4.3. Application of Reinforcement Learning in the Field of News Recommendation

Due to the highly dynamic nature of news features, user preferences, and the timeliness of news content, news recommendation remains a challenging and evolving area. Utilizing reinforcement learning not only enables the understanding of short-term and long-term changes in user interests but also facilitates the generation of personalized and targeted recommendations, aiming to maximize long-term rewards. Zheng et al. proposed DRN, a deep reinforcement learning framework based on Deep Q-Learning (DQN), to capture users' dynamic interests in news and predict future interactions between users and news. The DQN network is as shown in Fig 7.

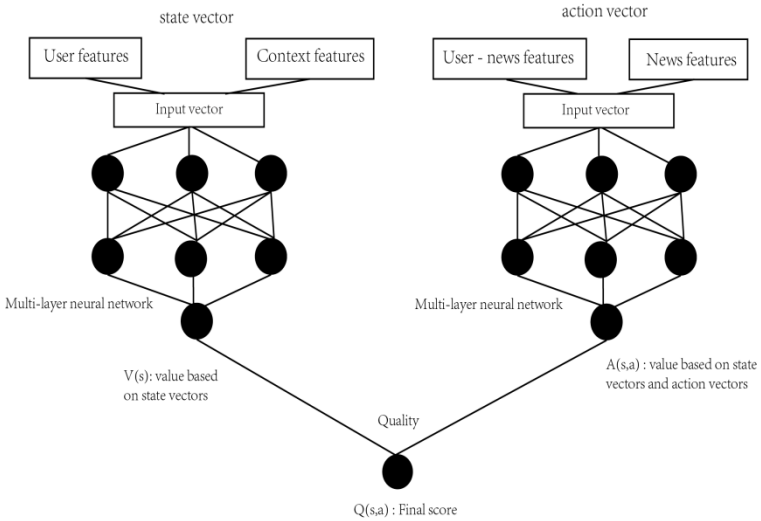


**Fig. 7** Framework of DQN (Photo/Picture credit: Original).

The model employs continuous state features representing user activity and continuous action features representing news items as inputs to a multi-layer deep Q-network. User activeness, serving as a supplement to Click Through Rate, is considered to account for user satisfaction after reading news, with both factors combined to calculate user feedback. The exploration algorithm incorporates Dueling Bandit gradient descent to enhance recommendation diversity, avoiding the detrimental

impact on recommendation accuracy caused by classical exploration strategies such as ϵ-greedy and Upper Confidence Bound. The model framework is outlined as shown in Fig 8:

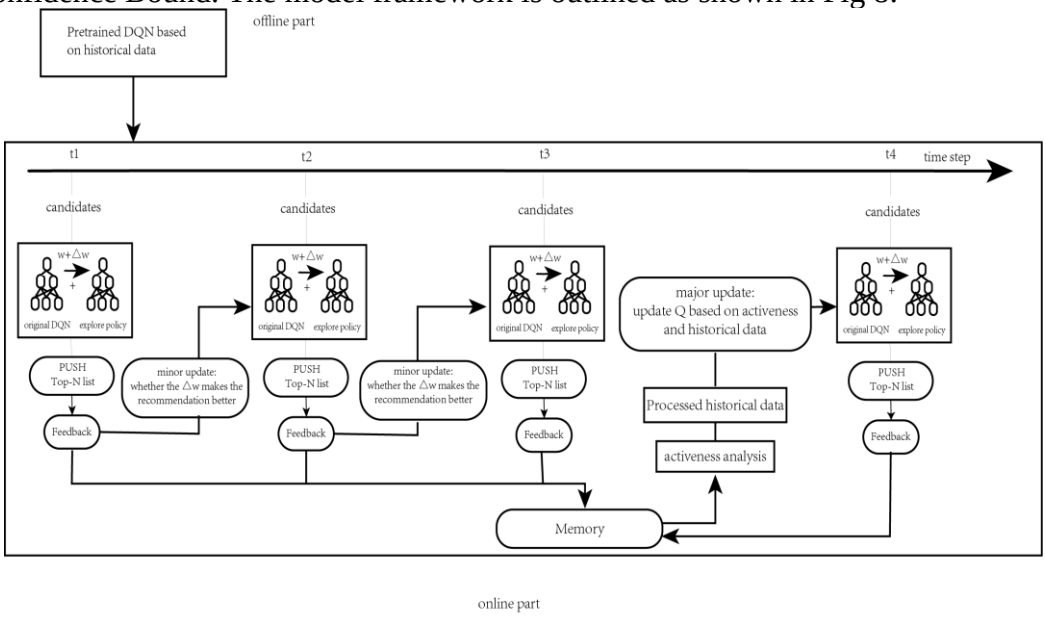


**Fig. 8** Framework of DRN (Photo/Picture credit: Original).

In the PUSH stage, at each time point ($t_i$=$t_1$, $t_2$, $t_3$, ...), the recommendation system takes the current user and candidate news features as input. Using the exploration mechanism of Deep Q-Regression, the system combine a recommendation list from original DQR with a newly generated recommendation list based on the offset produced by Dueling Bandit gradient descent ($\Delta W = a * rand(-1,1) * W$) and generates a recommended top-N news list, denoted as L. The user, upon receiving the recommended news in list L, provides a feedback. After each time ti, the system collects user feedback in real time. If the content generated by the exploration network is better than the current network, the exploration network replaces the current network for the next iteration; otherwise, the current network is retained. This gradient-descent-like approach keeps the model constantly synchronized with the most "fresh" data, integrating the latest feedback information into the model in real time.

After a certain period $T_R$ (here R is three), the agent G utilizes the experience replay technique, incorporating user feedback and activity stored in memory, to undergo a major update [36]. This major update involves updating the Q-network to enhance the overall learning and recommendation capabilities of the system.

## 5. Facing Challenges and Future Trends

### 5.1. Multi-Modal Recommendations

Due to the lack of multi-modal recommendation datasets, previous recommendation algorithms have predominantly worked on text datasets. However, users are often attracted by images, symbols, and video content, and it is challenging to accurately characterize user features solely based on text information. Therefore, incorporating multi-modal information into recommendation tasks is crucial to complement text content and achieve more accurate modeling in future research.

### 5.2. Fairer Recommendation Systems

Current recommendation systems often rely on explicit user actions such as clicks and shares. However, issues like "big data killing small businesses" and exposure biases are becoming more pronounced [37, 38]. Unfairness exists in the possession, distribution, and use of resources among social individuals or groups, significantly impacting user satisfaction, trust in algorithms, and

potentially leading to adverse social effects. For instance, in 2019, the U.S. Department of Housing and Urban Development sued Facebook for pushing ads based on attributes like gender, race, and religion. The algorithm, learned from historical data, suggested that showing real estate ads to men would generate more revenue compared to women, leading to gender discrimination [39]. Additionally, in news recommendations, titles containing discriminatory, violent, or radical terms may attract higher click rates, and measures should be considered in the reward mechanism of recommendation systems.

### 5.3. Establishing a More Reasonable Interactive Recommendation Evaluation Mechanism

Currently, scholars mostly use general metrics such as CTR, accuracy, recall, NDCG, MAP, etc., to evaluate reinforcement learning recommendation models. However, these metrics may not accurately reflect user satisfaction with the entire interaction process and the recommended results in interactive recommendation systems. These evaluation methods do not provide a genuine judgment of user satisfaction in specific application scenarios. For instance, in product recommendations, users often click on a product due to its image or price, but this may not represent the user's final satisfaction. Metrics such as adding to the shopping cart and product ratings are more critical. In news recommendations, users may randomly click on numerous headlines, and metrics like reading time and reading depth (whether they look for related news) serve as better evaluation indicators.

## 6. Conclusion

This comprehensive analysis elucidates the intricate dynamics of recommendation systems, highlighting the critical contribution of reinforcement learning techniques to their efficacy and user engagement. The pioneering DRN model by Zheng et al. and the innovative FeedRec model by Zou et al. exemplify this advancement. These models employ deep Q-networks and hierarchical LSTM networks, respectively, to dynamically adapt recommendations based on user activities and long-term satisfaction metrics. These approaches mark a significant progression in the field by adeptly integrating immediate user preferences with delayed feedback, utilizing advanced reward functions and matrix optimization techniques. The study's focus on coverage issues further accentuates the vital role of diversity in recommendations for enhancing user satisfaction. By differentiating between prediction and recommendation coverage, a more nuanced comprehension of how systems can present a wider array of choices to users is achieved, thereby enriching the user experience. Multi-Armed Bandit algorithms demonstrate their proficiency in striking a balance between exploration and exploitation, offering a mix of familiar and novel recommendations. Markov Decision Processes, with their emphasis on state transitions and policy-driven actions, along with Deep Reinforcement Learning, which applies deep learning techniques for optimizing values and policies, significantly refine the recommendation process. These methodologies underscore the evolving landscape of recommendation systems, where technological innovation meets user-centric design.

## References

[1] Marz, N., & Warren, J. (2015). Principles and best practices of scalable realtime data systems. James Warren, 328.

[2] Burke, R., Felfernig, A., & Göker, M. H. (2011). Recommender systems: An overview. AI Magazine, 32(3), 13-18.

[3] Linden, G., Smith, B., & York, J. (2003). Amazon.com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 7(1), 76-80.

[4] Cheng, H. T., Koc, L., Harmsen, J., Shaked, T., Chandra, T., Aradhye, H., ... & Shah, H. (2016, September). Wide & deep learning for recommender systems. In Proceedings of the 1st Workshop on Deep Learning for Recommender Systems (pp. 7-10).

[5] Covington, P., Adams, J., & Sargin, E. (2016, September). Deep neural networks for YouTube recommendations. In Proceedings of the 10th ACM Conference on Recommender Systems (pp. 191-198).

[6] Schuth, A., Oosterhuis, H., Whiteson, S., & De Rijke, M. (2016, February). Multileave gradient descent for fast online learning to rank. In Proceedings of the Ninth ACM International Conference on Web Search and Data Mining (pp. 457-466).

[7] Li, C., Feng, H., & De Rijke, M. (2020, September). Cascading hybrid bandits: Online learning to rank for relevance and diversity. In Proceedings of the 14th ACM Conference on Recommender Systems (pp. 33-42).

[8] Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). A brief survey of deep reinforcement learning. arXiv preprint arXiv:1708.05866.

[9] Pazahr, A., Zapater, J. J. S., Sánchez, F. G., Botella, C., & Martínez, R. (2016, November). Semantically-enhanced advertisement recommender systems in social networks. In Proceedings of the 18th International Conference on Information Integration and Web-based Applications and Services (pp. 179-189).

[10] Rohde, D., Bonner, S., Dunlop, T., Vasile, F., & Karatzoglou, A. (2018). Recogym: A reinforcement learning environment for the problem of product recommendation in online advertising. arXiv preprint arXiv:1808.00720.

[11] Zheng, G., Zhang, F., Zheng, Z., Xiang, Y., Yuan, N. J., Xie, X., & Li, Z. (2018, April). DRN: A deep reinforcement learning framework for news recommendation. In Proceedings of the 2018 world wide web conference (pp. 167-176).

[12] Elkahky, A. M., Song, Y., & He, X. (2015, May). A multi-view deep learning approach for cross domain user modeling in recommendation systems. In Proceedings of the 24th international conference on world wide web (pp. 278-288).

[13] Zhao, X., Xia, L., Zhang, L., Ding, Z., Yin, D., & Tang, J. (2018, September). Deep reinforcement learning for page-wise recommendations. In Proceedings of the 12th ACM conference on recommender systems (pp. 95-103).

[14] Zheng, L., Noroozi, V., & Yu, P. S. (2017, February). Joint deep modeling of users and items using reviews for recommendation. In Proceedings of the tenth ACM international conference on web search and data mining (pp. 425-434).

[15] Zhang, F., Yuan, N. J., Lian, D., Xie, X., & Ma, W. Y. (2016, August). Collaborative knowledge base embedding for recommender systems. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 353-362).

[16] Wang, H., Wang, N., & Yeung, D. Y. (2015, August). Collaborative deep learning for recommender systems. In Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1235-1244).

[17] Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. Communications of the ACM, 35(12), 61-70.

[18] Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th International Conference on World Wide Web (pp. 285-295).

[19] Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep learning based recommender system: A survey and new perspectives. ACM Computing Surveys (CSUR), 52(1), 1-38.

[20] Schuth, A., Oosterhuis, H., Whiteson, S., & de Rijke, M. (2016, February). Multileave gradient descent for fast online learning to rank. In proceedings of the ninth ACM international conference on web search and data mining (pp. 457-466).

[21] LEI Y, LI WJ. (2019) Interactive Recommendation with User-Specific Deep Reinforcement Learning. ACM Transactions on Knowledge Discovery from Data,13(6):11-15.

[22] Yang, L., Liu, B., Lin, L., ... & Chen, K., & Yang, Q. (2020, September). Exploring clustering of bandits for online recommendation system. In Proceedings of the 14th ACM Conference on Recommender Systems (pp. 120-129).

[23] Su, Y., ... & Xu, W. (2017, August). Improving click-through rate prediction accuracy in online advertising by transfer learning. In Proceedings of the International Conference on Web Intelligence (pp. 1018-1025).

[24] Al Qudah, D. A., Cristea, A. I., Bazdarevic, S. H., Al-Saqqa, S., Rodan, A., & Yang, W. (2015, October). Personalized e-advertisement and experience: recommending user targeted ads. In 2015 IEEE 12th International Conference on e-Business Engineering (pp. 56-61). IEEE.

[25] Lu, Y., Zhao, Z., Zhang, B., Ma, L., Huo, Y., & ...g, G. (2018). A context-aware budget-constrained targeted advertising system for vehicular networks. IEEE Access, 6, 8704-8713.

[26] Proios, D., Eirinaki, M., & Varlamis, I. (2015, August). TipMe: Personalized advertising and aspect-based opinion mining for users and businesses. In Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. 1489-1494.

[27] Zhao, X., Zhang, L., Ding, Z., ...a, L., Tang, J., & Yin, D. (2018, July). Recommendations with negative feedback via pairwise deep reinforcement learning. In Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (pp. 1040-1048).

[28] ZOU LX, XIA L, DING ZY, et al . Reinforcement Learning to Optimize Long-term User Engagement in Recommender Systems.

[29] Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS), 22(1), 5-53.

[30] Di Noia, T., Rosati, J., Tomeo, P., & Di Sciascio, E. (2017). Adaptive multi-attribute diversity for recommender systems. Information Sciences, 382, 234-253.

[31] Burke, R. (2002). Hybrid recommender systems: Survey and experiments. User modeling and user-adapted interaction, 12, 331-370.

[32] Weijun He, & Danxiang Ai. (2021). A multi-objective interactive recommendation system modeled as a multi-armed gambling machine. Journal of Chinese Computer Systems, 42(6), 1192-1198.

[33] Bray, B., & McClaskey, K. (2013). A Step-by-Step Guide to Personalize Learning. Learning & Leading with Technology, 40(7), 12-19.

[34] Tang, X., Chen, Y., Li, X., Liu, J., & Ying, Z. (2019). A reinforcement learning approach to personalized learning recommendation systems. British Journal of Mathematical and Statistical Psychology, 72(1), 108-135.

[35] Rupp, A. A., Templin, J., & Henson, R. A. (2010). Diagnostic measurement: Theory, methods, and applications. Guilford Press.

[36] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. (2015). Human-level control through deep reinforcement learning. Nature 518, 7540, 529–533.

[37] Gillis T B, Spiess J L. (2019) Big data and discrimination. The University of Chicago Law Review, 86(2): 459-488.

[38] Schmidt F. (2019) Generalization in generation: A closer look at exposure bias. arXiv preprint arXiv:1910.00292.

[39] Patil V, Ghalme G, Nair V, et al. (2021) Achieving fairness in the stochastic multi-armed bandit problem. The Journal of Machine Learning Research, 22(1): 7885-7915.