



Aggressive Exploration in Offline Reinforcement Learning for Better Recommendations

Kexin Shi¹, Wenjia Wang², and Bingyi Jing³

¹ The Hong Kong University of Science and Technology, Hong Kong SAR, China
kshiaf@connect.ust.hk

² The Hong Kong University of Science and Technology (Guangzhou), Guangzhou
511400, China wenjiawang@hkust-gz.edu.cn

³ Southern University of Science and Technology, Shenzhen 518000, China
jingby@sustech.edu.cn

Abstract. Offline reinforcement learning has become a powerful tool for optimizing recommender systems by learning from logged user interactions. However, existing methods rely on conservative exploration, limiting their ability to discover diverse and high-reward content. This paper introduces Bias-Reducing Aggressive Variance-Driven Exploration (BRAVE), an uncertainty-aware exploration strategy that effectively balances exploration and exploitation while addressing data bias to some extent in recommender systems. Unlike traditional offline RL methods that penalize uncertainty, BRAVE leverages uncertainty as a positive signal, guiding the agent toward underrepresented yet potentially high-reward recommendations. We evaluate BRAVE on KuaiRec, KuaiRand, and Yahoo datasets, demonstrating its effectiveness in prolonging user interaction and identifying highly relevant items, leading to improved user satisfaction. Moreover, BRAVE’s strong performance on biased datasets underscores the potential of aggressive exploration in offline RL, providing a novel approach to breaking filter bubbles and reducing bias in recommender systems.

Keywords: Recommender systems · Reinforcement learning · Uncertainty · Data bias.

1 Introduction

Recommender systems are central to many digital platforms, from e-commerce to entertainment, helping users discover content that aligns with their interests [12]. Despite their success, these systems often face challenges related to bias in the data and limited exploration of diverse content. A key issue is the filter bubble effect, where users are repeatedly exposed to similar content based on previous interactions, which can result in decreased long-term user satisfaction [10]. Therefore, improving recommendation systems requires overcoming biases and ensuring that recommendations better reflect the true preferences of users.

Reinforcement Learning (RL) [1] trains agents to make sequential decisions by maximizing cumulative rewards based on their interactions with the environment. In particular, offline RL has emerged as a powerful method for optimizing

recommender systems. It leverages historical interaction data to learn effective policies without the need for real-time user engagement, making it especially advantageous in scenarios where gathering immediate feedback is costly or impractical. The primary objective of offline RL is to enhance long-term user satisfaction by developing policies that effectively optimize user engagement and retention over time. Model-based RL has emerged as a promising approach in this field because of its sample efficiency [7]. By constructing a world model that simulates user-item interactions based on historical data, model-based RL allows the agent to predict the outcomes of different actions and plan accordingly. The accuracy of the world model is critical, as it determines how well the agent can generalize beyond the training data.

A significant challenge in offline RL for recommender systems is the bias inherent in logged data [3], such as exposure and selection biases, which makes it difficult to train models that accurately predict user preferences and evaluate recommender systems reliably in offline settings. The sparsity of data, where only a small fraction of possible user-item interactions are recorded, compounds this challenge. Another challenge is extrapolation error for offline RL. To mitigate extrapolation error, many offline RL methods adopt conservative strategy. They either constrain the policy to avoid selecting risky or out-of-distribution actions [30, 16, 6] or give a pessimistic estimate of the Q-function to account for uncertainty in the value of actions that have limited or no prior observations [29, 17, 15]. While this reduces extrapolation errors, it limits exploration, which can reinforce existing biases, restrict recommendation diversity, and enhance the filter bubble effect. Notably, model-based offline RL has the potential to reduce the need for conservative exploration. By improving the accuracy of the world model, exploration can be more effectively guided, even in sparse or biased datasets, allowing the system to better capture true user preferences and promote diverse recommendations without relying on overly cautious strategies.

This paper introduces **B**ias-**R**educing **A**ggressive **V**ariance-Driven **E**xploration (**BRAVE**), a model-based offline RL approach designed to enhance exploration in recommender systems by leveraging uncertainty, which represents the confidence in world model prediction outcomes. We show that uncertainty can serve as a valuable signal to distinguish between user-item pairs likely to generate positive feedback and those that are not. To leverage this insight, we propose a refined reward function that integrates uncertainty to promote guided “aggressive exploration”, directing the system to explore underutilized state-action pairs with higher uncertainty, rather than relying on random exploration. Compared to conventional conservative exploration strategies, aggressive exploration prevents premature convergence on suboptimal solutions and promotes greater diversity in recommendations, helping to break filter bubbles. Through extensive experiments on three datasets, our results show that BRAVE significantly enhances cumulative rewards, interaction length, and single-round reward. These findings demonstrate that our exploration strategy not only mitigates the biases inherent in offline data but also improves recommendation quality, showcasing

the potential of uncertainty-driven exploration to optimize recommender systems and increase user satisfaction. The contributions of this paper are:

- We analyze the impact of data bias on user-item predictions and enhance model robustness through uncertainty-based improvements for underrepresented interactions.
- We introduce BRAVE, an exploration strategy that leverages uncertainty to promote diverse recommendations, effectively breaking recommendation loops and reducing bias.
- Extensive experiments demonstrate that BRAVE outperforms baseline methods, achieving higher cumulative rewards, longer interaction lengths, and improved single-round rewards.

2 Related Work

Offline RL faces the challenge of extrapolation error, where policies may select actions outside the data distribution, resulting in unreliable outcomes. To mitigate this issue, many offline RL methods adopt conservative strategies. These strategies either constrain the policy to avoid risky or out-of-distribution actions [30, 16, 6] or provide pessimistic estimates of the Q-function to account for uncertainty in actions with limited or no prior observations [29, 17, 15]. For example, CQL [17] bounds the Q-function to avoid overestimation. BCQ [6] uses a generative model to restrict the action space to actions observed in the dataset. CRR [24] compares learned Q-values with observed ones, filtering out suboptimal actions and reducing deviations from the dataset.

While model-free offline RL methods mainly focus on regularizing the learned policy to prevent actions outside the observed data distribution, model-based offline RL methods lie in improving the world model’s accuracy to ensure more reliable decision-making from fixed datasets [13]. Meanwhile, a set of model-based offline RL approaches still incorporate conservative strategies like penalizing out-of-distribution state-action pairs [28, 27, 14]. For instance, MOPO [28] penalizes rewards based on model uncertainty, helping to minimize the influence of unreliable predictions. COMBO [27] regularizes the Q-function to penalize actions that fall outside the distribution of the training data.

Numerous studies have explored offline RL in recommender systems [7, 4, 33, 31, 34]. Nonetheless, biases in logged data and extrapolation errors still remain significant challenges in the domain of interactive recommendations [21, 3].

3 Method

3.1 Preliminaries

RL is a key area of artificial intelligence focused on developing optimal decision-making policies through interaction with an environment, represented by the Markov Decision Process (MDP) $\mathcal{M} = (S, A, T, r, \gamma)$. Here, S is the state space,

A is the action space, T defines transition dynamics, r is the reward function, and $\gamma \in (0, 1)$ is the discount factor prioritizing immediate rewards. The goal is to learn a policy $\pi(a|s)$ that maximizes the expected discounted return:

$$J(\pi) = \mathbb{E}_{\pi, T} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]. \quad (1)$$

Offline RL derives optimal policies from fixed datasets, $D_e = \{(s, a, r, s')\}$, collected from prior policies, but faces challenges like distributional shift, requiring effective generalization to unobserved states and actions. Model-based offline RL uses learned models of the environment’s dynamics to simulate interactions without real-time feedback. By estimating transition dynamics \hat{T} and reward function \hat{r} from D_e , it creates a new MDP, $\hat{\mathcal{M}} = (S, A, \hat{T}, \hat{r}, \gamma)$. While data-efficient, this approach relies heavily on the model’s generalization capability; poor generalization can lead to suboptimal decision-making and hinder real-world performance.

In recommendation tasks, an action $a \in A$ involves recommending an item i to a user based on a recommendation policy. The reward function $\hat{r} := \hat{r}(s, a)$ captures user feedback on action a conditioned on the user’s state s . This feedback can manifest in various forms, such as whether the user clicks on the item or the duration of engagement with content. The user state $s \in S$ reflects evolving preferences, but it is typically unobservable and must be inferred from past interactions using a user model. We formalize the user model as:

$$s_{t+1} = \text{User}(s_0, [a_1, a_2, \dots, a_{t+1}], [\hat{r}_1, \hat{r}_2, \dots, \hat{r}_{t+1}]), \quad (2)$$

where the User function estimates the state based on an initial state s_0 , a sequence of historical actions $[a_1, a_2, \dots, a_t]$, and their corresponding rewards $[\hat{r}_1, \hat{r}_2, \dots, \hat{r}_{t+1}]$. The initial state s_0 can be initialized using demographic or historical data, or randomly. In this framework, the transition dynamics \hat{T} and the reward function \hat{r} are estimated from a static dataset D_e . This enables the formulation of MDP.

3.2 World Model Study

In offline RL for recommender systems, it is crucial to construct a world model that simulates user state transitions and predicts the reward for each potential user–item interaction. While user–item interaction matrices form the typical training resource for such models, these matrices are frequently biased. Bias arises when recommendation policies selectively expose only certain items to users, yielding exposure bias, or when repeated exposure to popular items confines users to filter bubbles that reduce diversity and ultimately reduce long-term retention. Although a handful of unbiased datasets, such as Coat [22] and Yahoo! [19], have been collected by randomizing item exposure, they tend to be small and sparse, making them costly and less representative for large-scale training. Consequently, a key question is how to harness the abundant but biased

data effectively. Specifically, (1) how does the bias in partially observed matrices degrade the accuracy of user-item score predictions in the world model, and (2) how can we mitigate such bias in offline RL? To investigate these questions, we leverage datasets KuaiRec (biased) [8] and KuaiRand (unbiased) [9], both from the short video platform Kuaishou, to examine the impact of data bias on model performance and potential strategies for debiasing design.

World Model. We use DeepFM [11] to predict user-item interaction scores and generated user and item embeddings. Due to the partial observation of the training interaction matrix, some users and items had significantly more interaction logs than others, leading to an imbalance in prediction accuracy. To mitigate this issue and incorporate uncertainty, we modeled the predicted interaction scores \hat{y}_i with a Gaussian distribution [5]:

$$\hat{y}_i \sim \mathcal{N}(\mu_\theta(x_i), \sigma_\theta^2(x_i)), \quad (3)$$

where $\mu_\theta(x_i)$ is the predicted interaction score, and $\sigma_\theta^2(x_i)$ represents the variance, capturing the uncertainty in the prediction. The world model is trained by minimizing a negative log-likelihood loss function:

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^N \left(\frac{1}{2\sigma_\theta^2(x_i)} \|y_i - \mu_\theta(x_i)\|^2 + \frac{1}{2} \log \sigma_\theta^2(x_i) \right), \quad (4)$$

where N is the total number of observed interactions, y_i is the ground truth interaction score, $\mu_\theta(x_i)$ and $\sigma_\theta^2(x_i)$ are the predicted score and variance for sample x_i , respectively.

By incorporating this uncertainty-aware training objective, the model not only predicts accurate interaction scores for each user-item pair but also quantifies its confidence through the predicted variances.

To further improve reliability, we adopt an ensemble of world models. The final predictions for the scores are obtained by averaging the predicted scores across the ensemble, while the variance for each sample x_i is taken as the maximum value among the predicted variances from the ensemble:

$$\mu_\theta(x_i) = \frac{1}{M} \sum_{m=1}^M \mu_\theta^{(m)}(x_i), \quad \sigma_\theta^2(x_i) = \max_{m=1, \dots, M} \sigma_\theta^{2(m)}(x_i), \quad (5)$$

where M is the number of models in the ensemble, and $\mu_\theta^{(m)}(x_i)$ and $\sigma_\theta^{2(m)}(x_i)$ are the score and variance predictions from the m -th model, respectively.

World Model Performance. We evaluate the ability of the trained world model to differentiate between positive and negative items within the evaluation matrix.

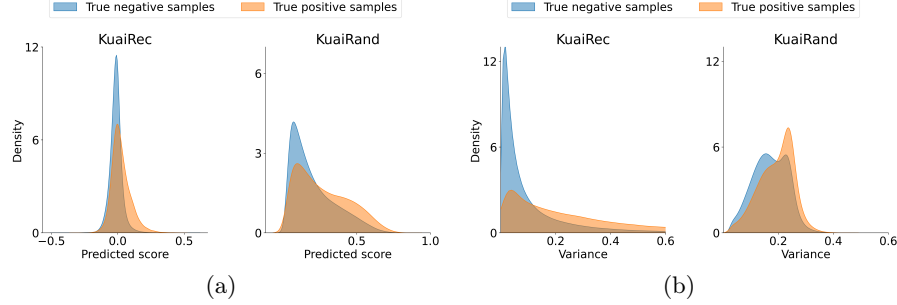


Fig. 1. Distribution of predicted scores and variance for true positive and true negative samples.

Differentiating Positive and Negative Samples. To assess the model’s performance, we analyze the normalized predicted scores x_i for both positive and negative samples. The score distributions for these categories across the KuaiRec (biased) and KuaiRand (unbiased) datasets are shown in Figure 1(a). Our analysis reveals that the predicted scores for positive samples are skewed towards higher values, indicating that the world model effectively captures user preferences:

$$\mu_{\theta}(x_i|\text{positive}) > \mu_{\theta}(x_i|\text{negative}). \quad (6)$$

Examining Uncertainty in Predictions. We also examine the model’s uncertainty measure, specifically the predicted variance $\sigma_{\theta}^2(x_i)$, for both positive and negative samples. The distributions of these variances, illustrated in Figure 1(b), show that positive samples tend to have higher variance than negative samples:

$$\sigma_{\theta}^2(x_i|\text{positive}) > \sigma_{\theta}^2(x_i|\text{negative}). \quad (7)$$

This suggests that the model is more uncertain when predicting scores for positive samples. This phenomenon may arise because many user-item pairs in the training data lack interactions or are associated with low watch times. As a result, the model becomes more confident in identifying negative samples, while the limited number of positive samples during training leads to increased prediction uncertainty. These findings imply that the uncertainty measure $\sigma_{\theta}^2(x_i)$ could be used as an additional signal to classify positive and negative samples, beyond relying solely on the predicted scores.

The Effect of Biased Training Data on Uncertainty. We observe significant differences in the variance distributions between KuaiRec and KuaiRand. In KuaiRec, the variance distribution for positive samples exhibits a heavy tail, indicating that some positive samples are associated with substantially higher uncertainty. In contrast, the distributions for positive and negative samples in KuaiRand are more similar, with no heavy tail present. The discrepancy may arise from the biased nature of KuaiRec, which predominantly includes items recommended

based on personal preference analysis and popular items with high interaction frequencies. This bias leads to fewer diverse examples of positive user-item interactions in the training data, limiting the model’s ability to confidently predict less common positive samples. Consequently, the lack of diverse training samples for positive interactions contributes to heightened uncertainty in the model’s predictions, as evidenced by the heavy-tailed variance distribution in KuaiRec.

Because our analysis of datasets within recommendation systems is broadly applicable, these phenomena still occur when the world model is altered, rather than being incidental to a specific model.

3.3 Reward Enhancement

Building on the insights gained from the world model, we observed that the variance generated for each sample reflects the model’s confidence in its predictions and provides valuable information about the underlying data distribution. Notably, potential positives tend to exhibit comparatively larger predicted uncertainty than negatives. This characteristic is likely tied to both the sparsity and the bias inherent in the training data, as well as domain-specific dynamics in recommender systems.

Inspired by the findings, we propose an enhanced reward function that incorporates predicted variance to promote exploration in offline RL. The enhanced reward function is defined as:

$$\tilde{r}(s, a) = \hat{r}(s, a) + \lambda U(s, a), \quad (8)$$

where $\hat{r}(s, a)$ is the predicted reward from the world model, $U(s, a)$ represents the variance derived from the Gaussian prediction of the world model, capturing the uncertainty associated with the sample and λ is a scaling factor controlling the contribution of the variance term. This function not only captures user preferences through the predicted reward but also incentivizes the agent to explore state-action pairs with higher uncertainty, which may reveal potential positives.

Exploration-Exploitation Balance and Bias Mitigation. The enhanced reward function combines the predicted reward ($\hat{r}(s, a)$) with the variance-driven term (λU), effectively balancing the exploitation of promising user preferences with the exploration of uncertain, potentially positive interactions. By integrating both factors, the reward function focuses the RL agent’s attention on interactions that not only exhibit strong indicators of user preference but also hold the potential for discovery in regions of the state-action space marked by uncertainty.

This design is effective for both biased and unbiased datasets, as the distribution of predicted variance for positive samples consistently shifts to the right compared to that of negative samples. However, it is particularly impactful for biased datasets, where frequently interacted or popular items dominate the training data, reinforcing narrow recommendation loops. In such datasets, the

distribution of predicted variance for positives not only shifts but also exhibits a heavy tail—a distinct shape compared to that of negatives. This heavy-tailed distribution provides a critical signal for distinguishing positives from negatives, especially for underrepresented interactions. By leveraging this variance signal, the enhanced reward function is expected to break the cycle of bias by guiding the RL agent to explore diverse interactions, including those with high promise but limited representation in the data. This approach ensures that the model considers a broader range of interactions, promoting diversity and fairness in recommendations while still maintaining strong alignment with user preferences.

Learning Pipeline. In this paper, we adopt the experimental setup of DORL [7], with our primary contribution being the redesign of the reward function informed by our empirical study of the world model in recommender systems.

We employ DeepFM as the underlying model to estimate the user state s . DeepFM predicts entries in the user-item interaction matrix as reward signals \hat{r} , while simultaneously generating user embeddings e_u and item embeddings e_i . The evolution of the user state is achieved by dynamically integrating recent actions and their feedback, ensuring that the representation adapts to both immediate user reactions and long-term behavioral trends. At timestamp t , the updated user state s_{t+1} is computed as:

$$s_{t+1} = \frac{1}{N} \sum_{k=t-N+1}^t [e_{a_k} \oplus \hat{r}(s_k, a_k)], \quad (9)$$

where N denotes the number of most recent actions considered for the state update. Here, e_{a_k} corresponds to the embedding of action a_k , capturing its latent features, while $\hat{r}(s_k, a_k)$ represents the predicted reward for action a_k , conditioned on the prior state s_k . The operation \oplus combines the action embedding with the predicted reward, yielding a unified representation that integrates user feedback. Advantage Actor-Critic [20] (A2C) algorithm is adopted to train the recommendation policy, leveraging its ability to model dynamic user preferences and adapt to sequential decision-making.

3.4 Conservative Strategy vs. Aggressive Strategy

Exploration in model-based offline RL plays a crucial role in balancing the trade-off between exploiting known high-reward actions and discovering new, potentially optimal actions. In practice, most existing methods adopt conservative exploration strategies, designed to mitigate risks associated with OOD actions. However, we argue that aggressive exploration can be more suitable for recommendation systems due to their distinct dynamics, such as users’ desire for diversity and tolerance for novelty.

Conservative strategies are widely used in high-stakes environments like robotics, where trial-and-error can lead to catastrophic failures. Offline RL methods, whether model-based or model-free, tend to exhibit conservative behaviors.

Model-free methods either constrain the policy to avoid selecting risky or out-of-distribution actions or give a pessimistic estimate of the Q-function. Model-based offline RL methods learn from historical data, but the limited and unrepresentative nature of this data introduces uncertainty, leading many methods to adopt pessimistic policies to avoid high-risk actions. For instance, MOPO (Model-Based Offline Policy Optimization) employs a conservative reward function defined as:

$$\hat{r}(s, a) = r(s, a) - \lambda U(s, a), \quad (10)$$

where $r(s, a)$ is the predicted reward from the learned reward model and $U(s, a)$ represents the uncertainty of that prediction. This formulation penalizes actions that lead to highly uncertain outcomes, ensuring that the policy remains within the offline data distribution. MOPO has demonstrated effectiveness in domains such as robotics, where safety and reliability are critical.

In recommender systems, many studies adopt the concept of penalizing uncertainty from general offline RL algorithms. For example, Gao et al. proposed the following reward function [7]:

$$\hat{r}(s, a) = r(s, a) - \lambda_U U(s, a) + \lambda_E P_E(s), \quad (11)$$

Similarly, Zhang et al. developed a refined reward function [32]:

$$\hat{r}(s, a) = \tilde{r}(s, a) \times (1 - \tilde{U}(s, a)) + \lambda_E P_E, \quad (12)$$

In both modified reward functions, uncertainty ($U(s, a)$ or $\tilde{U}(s, a)$) is penalized, although these works introduce entropy-based penalties $P_E(s)$ to promote diversity. These methods aim to increase diversity on one hand, while being reluctant to abandon conservatism on the other, resulting in opposing effects.

Aggressive Exploration in Recommendation Systems. While conservative strategies are effective in high-risk domains like healthcare, autonomous driving, and finance, such conservatism can be overly restrictive in recommender systems. In recommender systems, where the cost of exploration is relatively low, users actively seek diverse and novel content. A conservative approach that focuses primarily on exploiting past behaviors and known preferences often limits the discovery of new and unexpected content, which can lead to stagnation and reduced user engagement.

Several studies highlight the adverse effects of homogeneous recommendations. Specifically, users become dissatisfied when they are repeatedly exposed to similar content, ultimately leading to decreased system usage and reduced satisfaction [2]. Furthermore, recommending overly familiar items, while ensuring relevance, fails to foster long-term engagement because users do not experience novelty or serendipity [26]. Consequently, an exclusive reliance on conservative strategies may hinder the system’s ability to discover optimal solutions within unexplored areas of the content space. In contrast to previous methods, our approach embraces uncertainty rather than penalizing it, treating it as a signal to encourage exploration. This is the core ingredient of our exploration strategy.

Additionally, unlike methods that promote action diversity at the policy level, our exploration is directly tied to the world model, leveraging the structure of the offline data to guide exploration more effectively. By aligning exploration with data-informed insights, our method enables the discovery of high-reward actions that conservative strategies often overlook.

4 Experiments

4.1 Experimental Setup

In this study, we evaluate our proposed approach using three widely recognized recommendation datasets: KuaiRec [8], which features a biased training set with a fully observed evaluation matrix; KuaiRand [9], which consists of both unbiased training and evaluation sets; and Yahoo [19], characterized by biased training data with a randomly sampled evaluation set.

For our evaluation, we utilize three key metrics: Cumulative Reward (R_{tra}), which represents the total rewards accumulated during an interaction session; Interaction Length (Length), defined as the number of consecutive recommendations made before the termination of a user session; and Single-Round Reward (R_{each}), which reflects the average reward obtained from a single recommendation step. To simulate user interaction termination effectively, we implement a quit mechanism consistent with methodologies employed in prior research [7, 32, 10]. This quit mechanism is easily triggered when the system repeatedly recommends items from the same category to a user, reinforcing the need for diversity in recommendations, as excessive familiarity of items can diminish user satisfaction and engagement [2, 26].

We compare our method against a range of baseline approaches, including bandit algorithms (ϵ -Greedy and Upper Confidence Bound (UCB) [18]), as well as model-free methods like SQN [25], BCQ [6], CQL [17], and CRR [24]. Additionally, we assess model-based offline RL techniques, including IPS [23], MBPO [13], MOPO [28], and DORL [7]. Comprehensive details regarding the datasets, implementation specifics, and baseline methodologies are available in the supplementary materials.

4.2 Overall Performance

The experimental results are shown in Table 1, and the corresponding training curves are presented in Figure 2.

In terms of cumulative reward, the most crucial performance metric that reflects long-term user satisfaction and engagement, our approach significantly outperforms all baseline methods across three datasets. Specifically, BRAVE achieves a relative improvement of 38.2% over the best baseline on KuaiRec, 12.0% on KuaiRand and 3.90% on Yahoo.

For KuaiRec, both BRAVE and DORL achieve an interaction length above 26, while all other baselines fall below 17. Similarly, for KuaiRand, BRAVE and

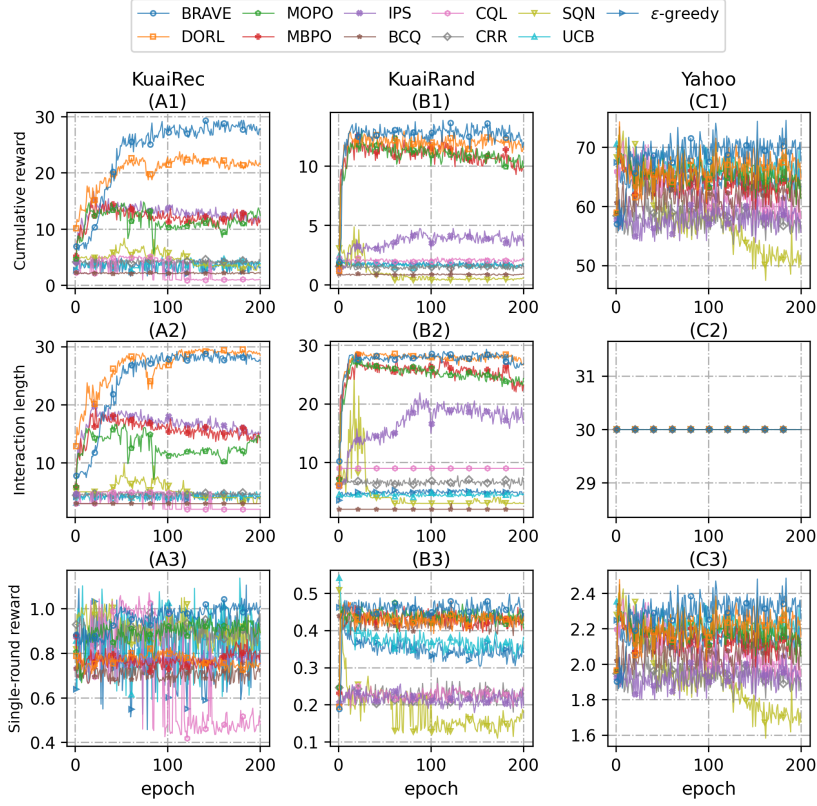


Fig. 2. Training curves for R_{tra} , R_{each} and Length.

DORL show approximately a 6% improvement in interaction length over other methods. (In Yahoo, items are finely categorized, leading to very few overlaps. This makes it challenging to trigger the quit mechanism in the experimental setting. Thus, the maximum value we set for the interactive environment can be easily reached across different methods.) Importantly, BRAVE outperforms DORL in terms of single-round reward, with relative improvements of 24.3% on KuaiRec, 12.9% on KuaiRand and 5.70% on Yahoo. These gains highlight BRAVE’s ability to learn from logged data while uncovering users’ true preferences.

In addition, the performance of the five model-based offline RL approaches (BRAVE, DORL, MBPO, MOPO, and IPS) is superior to that of model-free methods and bandit methods for KuaiRec and KuaiRand. This may be attributed to the sparsity of training data in recommender systems, where model-based approaches can better leverage the limited interactions by using learned models to simulate missing data. In contrast, model-free methods, which rely

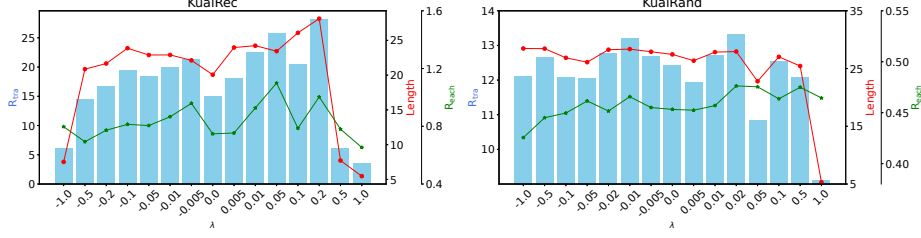


Fig. 3. Impact of hyperparameter λ on R_{tra} , R_{reach} and Length.

Table 1. Performance comparison on datasets KuaiRec, KuaiRand and Yahoo. (The best results are indicated in bold, and the second-best results are underlined.)

Methods	KuaiRec			KuaiRand		
	R_{tra}	R_{reach}	Length	R_{tra}	R_{reach}	Length
UCB	3.606 ± 0.609	0.853 ± 0.114	4.219 ± 0.389	1.651 ± 0.152	0.372 ± 0.028	4.431 ± 0.212
ϵ -greedy	3.515 ± 0.731	0.828 ± 0.129	4.219 ± 0.405	1.711 ± 0.126	0.351 ± 0.025	4.880 ± 0.270
SQN	4.673 ± 1.215	<u>0.913 ± 0.055</u>	5.111 ± 1.288	0.912 ± 0.929	0.182 ± 0.058	4.601 ± 3.712
CRR	4.163 ± 0.253	0.895 ± 0.037	4.654 ± 0.215	1.481 ± 0.124	0.226 ± 0.015	6.561 ± 0.352
CQL	2.506 ± 1.767	0.684 ± 0.228	3.224 ± 1.365	2.032 ± 0.107	0.226 ± 0.012	9.000 ± 0.000
BCQ	2.123 ± 0.081	0.708 ± 0.027	3.000 ± 0.000	0.852 ± 0.052	0.425 ± 0.016	2.005 ± 0.071
MBPO	12.043 ± 1.312	0.770 ± 0.029	15.646 ± 1.637	10.933 ± 0.946	0.431 ± 0.021	25.345 ± 1.819
IPS	12.833 ± 1.353	0.767 ± 0.023	16.727 ± 1.683	3.629 ± 0.676	0.216 ± 0.014	16.821 ± 3.182
MOPO	11.427 ± 1.750	0.892 ± 0.051	12.809 ± 1.850	10.934 ± 0.963	<u>0.437 ± 0.019</u>	25.002 ± 1.891
DORL	20.494 ± 2.671	0.767 ± 0.026	26.712 ± 3.419	11.850 ± 1.036	0.428 ± 0.022	27.609 ± 2.121
Ours	28.328 ± 2.052	0.953 ± 0.063	28.010 ± 1.072	13.277 ± 0.960	0.483 ± 0.021	26.860 ± 2.351

Methods	Yahoo		
	R_{tra}	R_{reach}	Length
UCB	66.758 ± 1.254	2.225 ± 0.042	30.000 ± 0.000
ϵ -greedy	64.344 ± 1.291	2.145 ± 0.043	30.000 ± 0.000
SQN	57.727 ± 5.751	1.924 ± 0.192	30.000 ± 0.000
CRR	57.994 ± 1.675	1.933 ± 0.056	30.000 ± 0.000
CQL	62.291 ± 3.347	2.076 ± 0.112	30.000 ± 0.000
BCQ	61.739 ± 1.781	2.058 ± 0.059	30.000 ± 0.000
MBPO	64.550 ± 2.157	2.152 ± 0.072	30.000 ± 0.000
IPS	57.850 ± 1.796	1.928 ± 0.060	30.000 ± 0.000
MOPO	65.510 ± 2.100	2.184 ± 0.070	30.000 ± 0.000
DORL	66.351 ± 2.224	2.212 ± 0.074	30.000 ± 0.000
Ours	69.360 ± 1.362	2.338 ± 0.0637	30.000 ± 0.000

directly on the observed data, may struggle to generalize effectively from sparse interactions.

4.3 Debiasing Ability

In Figure 2, we observe that the top-performing methods, including MOPO, MBPO, DORL, BRAVE, and IPS, exhibit significant performance variation on the biased KuaiRec dataset compared to the unbiased KuaiRand dataset. This variation likely stems from their interaction with exposure bias. (Yahoo differs from KuaiRec and KuaiRand by originating from a different platform and showing minimal performance variation across all methods, suggesting it is easier to learn from. Thus, we exclude it from the following discussion.)

For KuaiRec, exposure bias is inherent, meaning that certain items are more likely to be shown to users based on previous recommendations or popularity, which creates a skewed distribution of item interactions. This leads to an over-representation of certain items, resulting in a reduced diversity of interactions. Methods such as BRAVE and DORL, which balance exploration and exploitation—where DORL enhances policy entropy as outlined in Eq. 11—can explore diverse options and break the recommendation loops caused by the exposure bias. This can lead to significant improvements in interaction length by offering a broader selection of items, which helps mitigate the filter bubble. In contrast, interaction logs in KuaiRand are gathered by randomly inserting randomly selected videos into users’ recommendation streams. This random exposure strategy ensures unbiased interaction data. Since there is less skew in the interactions, methods like BRAVE and DORL do not experience as much of an advantage from their exploration strategies. The improvements are relatively smaller because there is already a good balance of diversity in the logged data.

This explains why performance differences between methods are more pronounced in biased dataset KuaiRec but less significant in unbiased dataset KuaiRand, highlighting the importance of how exploration and exploitation strategies interact with data bias in recommendation systems.

4.4 Comparison among MOPO, MBPO and BRAVE

We compare three methods—MOPO, MBPO, and BRAVE—which all use the same world model but differ in how they handle uncertainty: MOPO penalizes uncertainty, MBPO ignores it, and BRAVE encourages it. On three datasets, BRAVE significantly outperforms both MOPO and MBPO in terms of cumulative reward, highlighting the potential of aggressive exploration strategies in recommendation systems. BRAVE achieves a larger interaction length (on KuaiRec and KuaiRand) and higher single-round reward, which both account for its highest cumulative reward. The increased interaction length can be attributed to BRAVE’s ability to explore a wider range of item possibilities. This strategy is particularly effective at breaking the filter bubble. In our experimental setup, this exploration leads to improved user retention, as users are exposed to more diverse items and engage with the platform for longer periods. The higher single-round reward is due to BRAVE using uncertainty as an additional signal for distinguishing true positive and negative samples, improving prediction accuracy for unseen items, as suggested in section 3.2.

In addition, MOPO, which employs a more conservative exploration strategy by penalizing uncertainty, performs better than MBPO in terms of single-round reward. This is likely because penalizing uncertainty helps improve the prediction accuracy for items that occur more frequently in the training data. By reducing the model’s exploration of uncertain states, MOPO effectively improves the reliability of its recommendations for well-represented items in the training set. However, this approach limits exploration of items that the model is less confident about (i.e., unseen or infrequent items), which are typically the ones that could diversify user experiences and break the filter bubble. Thus, MOPO’s

conservative strategy often results in shorter interaction lengths compared to MBPO.

4.5 Analysis of λ Impact on Performance

As shown in Figure 3, the impact of the hyperparameter λ , which controls the contribution of variance (uncertainty) to the reward function, is analyzed on both KuaiRec and KuaiRand environments. When $\lambda < 0$, the system penalizes uncertainty, following a more conservative exploration strategy. Conversely, $\lambda > 0$ means employing an aggressive exploration strategy.

For the biased dataset KuaiRec, BRAVE shows substantial improvement as λ increases, reaching its peak at $\lambda = 0.2$. It demonstrates the benefits of aggressive exploration compared to the conservative approach. This results in higher cumulative rewards and longer interaction lengths. On the other hand, for KuaiRand (unbiased), the model’s performance shows less variation across different λ values, but the best performance is still achieved with aggressive exploration at $\lambda = 0.02$. Furthermore, a general upward trend in single-round reward is observed with positive values of λ , compared to negative values, across both datasets. This highlights BRAVE’s ability to more effectively predict and uncover users’ latent preferences, providing higher-quality recommendations in dynamic interactive environments.

5 Conclusion

This paper introduced BRAVE, an aggressive exploration strategy for offline RL in recommender systems. Inspired by the world model study, BRAVE incorporates uncertainty into the reward function to enhance the prediction of true positive and negative items. Unlike prevalent conservative exploration strategies, BRAVE enables a more effective balance between exploration and exploitation, ultimately improving cumulative rewards. This exploration method encourages the model to search for the global optimum, rather than being confined to the recommendation loops created by the recommendation policy. Through extensive experiments on datasets KuaiRec, KuaiRand and Yahoo, we demonstrated that BRAVE outperforms baseline methods, uncovering true user preferences and providing more diverse and relevant recommendations. BRAVE shows significant potential for addressing the challenges of bias in offline RL and future work will focus on refining the exploration strategies further and exploring real-world applications of BRAVE.

6 Acknowledgements

This work was partially supported by NSFC under grant number 12371290, for which we are sincerely grateful. The authors would also like to extend their heartfelt thanks to the editor and the reviewers for their valuable feedback and insightful comments.

References

1. Barto, A.G.: Reinforcement learning: An introduction. by richard's sutton. SIAM Rev **6**(2), 423 (2021)
2. Chaney, A.J., Stewart, B.M., Engelhardt, B.E.: How algorithmic confounding in recommendation systems increases homogeneity and decreases utility. In: Proceedings of the 12th ACM conference on recommender systems. pp. 224–232 (2018)
3. Chen, J., Dong, H., Wang, X., Feng, F., Wang, M., He, X.: Bias and debias in recommender system: A survey and future directions. ACM Transactions on Information Systems **41**(3), 1–39 (2023)
4. Chen, M., Beutel, A., Covington, P., Jain, S., Belletti, F., Chi, E.H.: Top-k off-policy correction for a reinforce recommender system. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. pp. 456–464 (2019)
5. Deisenroth, M., Rasmussen, C.E.: Pilco: A model-based and data-efficient approach to policy search. In: Proceedings of the 28th International Conference on machine learning (ICML-11). pp. 465–472 (2011)
6. Fujimoto, S., Meger, D., Precup, D.: Off-policy deep reinforcement learning without exploration. In: International conference on machine learning. pp. 2052–2062. PMLR (2019)
7. Gao, C., Huang, K., Chen, J., Zhang, Y., Li, B., Jiang, P., Wang, S., Zhang, Z., He, X.: Alleviating matthew effect of offline reinforcement learning in interactive recommendation. In: Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval. pp. 238–248 (2023)
8. Gao, C., Li, S., Lei, W., Chen, J., Li, B., Jiang, P., He, X., Mao, J., Chua, T.S.: Kuairc: A fully-observed dataset and insights for evaluating recommender systems. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. pp. 540–550 (2022)
9. Gao, C., Li, S., Zhang, Y., Chen, J., Li, B., Lei, W., Jiang, P., He, X.: Kuairand: An unbiased sequential recommendation dataset with randomly exposed videos. In: Proceedings of the 31st ACM International Conference on Information & Knowledge Management. pp. 3953–3957 (2022)
10. Gao, C., Wang, S., Li, S., Chen, J., He, X., Lei, W., Li, B., Zhang, Y., Jiang, P.: Cirs: Bursting filter bubbles by counterfactual interactive recommender system. ACM Transactions on Information Systems **42**(1), 1–27 (2023)
11. Guo, H., Tang, R., Ye, Y., Li, Z., He, X.: Deepfm: a factorization-machine based neural network for ctr prediction. arXiv preprint arXiv:1703.04247 (2017)
12. Hu, Y., Koren, Y., Volinsky, C.: Collaborative filtering for implicit feedback datasets. In: 2008 Eighth IEEE international conference on data mining. pp. 263–272. Ieee (2008)
13. Janner, M., Fu, J., Zhang, M., Levine, S.: When to trust your model: Model-based policy optimization. Advances in neural information processing systems **32** (2019)
14. Kidambi, R., Rajeswaran, A., Netrapalli, P., Joachims, T.: Morel: Model-based offline reinforcement learning. Advances in neural information processing systems **33**, 21810–21823 (2020)
15. Kostrikov, I., Nair, A., Levine, S.: Offline reinforcement learning with implicit q-learning. arXiv preprint arXiv:2110.06169 (2021)
16. Kumar, A., Fu, J., Soh, M., Tucker, G., Levine, S.: Stabilizing off-policy q-learning via bootstrapping error reduction. Advances in neural information processing systems **32** (2019)

17. Kumar, A., Zhou, A., Tucker, G., Levine, S.: Conservative q-learning for offline reinforcement learning. *Advances in neural information processing systems* **33**, 1179–1191 (2020)
18. Lai, T.L., Robbins, H.: Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics* **6**(1), 4–22 (1985)
19. Marlin, B.M., Zemel, R.S.: Collaborative prediction and ranking with non-random missing data. In: *Proceedings of the third ACM conference on Recommender systems*. pp. 5–12 (2009)
20. Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., Kavukcuoglu, K.: Asynchronous methods for deep reinforcement learning. In: *International conference on machine learning*. pp. 1928–1937. PmLR (2016)
21. Pradel, B., Usunier, N., Gallinari, P.: Ranking with non-random missing ratings: influence of popularity and positivity on evaluation metrics. In: *Proceedings of the sixth ACM conference on Recommender systems*. pp. 147–154 (2012)
22. Schnabel, T., Swaminathan, A., Singh, A., Chandak, N., Joachims, T.: Recommendations as treatments: Debiasing learning and evaluation. In: *international conference on machine learning*. pp. 1670–1679. PMLR (2016)
23. Swaminathan, A., Joachims, T.: Counterfactual risk minimization: Learning from logged bandit feedback. In: *International conference on machine learning*. pp. 814–823. PMLR (2015)
24. Wang, Z., Novikov, A., Zolna, K., Merel, J.S., Springenberg, J.T., Reed, S.E., Shahriari, B., Siegel, N., Gulcehre, C., Heess, N., et al.: Critic regularized regression. *Advances in Neural Information Processing Systems* **33**, 7768–7778 (2020)
25. Xin, X., Karatzoglou, A., Arapakis, I., Jose, J.M.: Self-supervised reinforcement learning for recommender systems. In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. pp. 931–940 (2020)
26. Yu, C., Lakshmanan, L., Amer-Yahia, S.: It takes variety to make a world: diversification in recommender systems. In: *Proceedings of the 12th international conference on extending database technology: Advances in database technology*. pp. 368–378 (2009)
27. Yu, T., Kumar, A., Rafailov, R., Rajeswaran, A., Levine, S., Finn, C.: Combo: Conservative offline model-based policy optimization. *Advances in neural information processing systems* **34**, 28954–28967 (2021)
28. Yu, T., Thomas, G., Yu, L., Ermon, S., Zou, J.Y., Levine, S., Finn, C., Ma, T.: Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems* **33**, 14129–14142 (2020)
29. Zhang, J., Fang, L., Shi, K., Wang, W., Jing, B.: Q-distribution guided q-learning for offline reinforcement learning: Uncertainty penalized q-value via consistency model. *Advances in Neural Information Processing Systems* **37**, 54421–54462 (2025)
30. Zhang, J., Zhang, C., Wang, W., Jing, B.: Constrained policy optimization with explicit behavior density for offline reinforcement learning. *Advances in Neural Information Processing Systems* **36**, 5616–5630 (2023)
31. Zhang, R., Yu, T., Shen, Y., Jin, H., Chen, C., Carin, L.: Reward constrained interactive recommendation with natural language feedback. *arXiv preprint arXiv:2005.01618* (2020)
32. Zhang, Y., Qiu, R., Liu, J., Wang, S.: Roler: Effective reward shaping in offline reinforcement learning for recommender systems. In: *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*. pp. 3269–3278 (2024)

33. Zhao, X., Xia, L., Zhang, L., Ding, Z., Yin, D., Tang, J.: Deep reinforcement learning for page-wise recommendations. In: Proceedings of the 12th ACM conference on recommender systems. pp. 95–103 (2018)
34. Zhao, X., Xia, L., Zou, L., Liu, H., Yin, D., Tang, J.: Whole-chain recommendations. In: Proceedings of the 29th ACM international conference on information & knowledge management. pp. 1883–1891 (2020)