ACPR: Proactive Recommendation under User Acceptance Constraints

Yexuan Che, Lei Zhang*, Siyue Sheng, Zhendong Mao, Yongdong Zhang
University of Science and Technology of China
Hefei, China
yexuanche@mail.ustc.edu.cn, leizh23@ustc.edu.cn, shengsiyue1@mail.ustc.edu.cn,
zdmao@ustc.edu.cn, zhyd73@ustc.edu.cn

Abstract*—Recommender systems typically prioritize immediate satisfaction, inadvertently reinforcing the Filter Bubble and limiting exploration. Proactive Recommendation addresses this by enhancing interest in a target item through a sequence of intermediate items. Prior works typically assume that users will passively accept all intermediate items, and primarily focus on the interest uplift toward the target item. However, in practice, if an intermediate item deviates significantly from a user's interest, the user is likely to actively abandon the current session, resulting in the complete failure of the target item's exposure. To address this, we propose the Acceptance-Constrained Proactive Recommendation (ACPR) framework, formulating proactive recommendation as a Constrained Markov Decision Process (CMDP) with user acceptance as a safety constraint. Instead of optimizing cumulative rewards related to the target, ACPR employs a step-wise greedy strategy to maintain every intermediate item within the user's current acceptance region, securing a feasible sequence for effective target exposure. Specifically, we instantiate the policy using a Large Language Model (LLM) to leverage its rich semantic knowledge to infer natural and logical connections between diverse items. To solve the constrained optimization problem, we integrate Reward Constrained Policy Optimization (RCPO) with Group Relative Policy Optimization (GRPO) to balance the proactive objective and acceptance constraint. Extensive experiments demonstrate that ACPR effectively improves users' interest in target items while maintaining high intermediate acceptance compared to state-of-the-art methods. The advantage of ACPR is particularly evident in scenarios with larger interest gaps, especially in sparse data scenarios where limited user history makes preference understanding more difficult. Code is available at: https://github.com/YexuanChe00/ACPR

***Index Terms*—Proactive Recommendation, Large Language Models, Constrained Reinforcement Learning**

I. Introduction

Recommender systems typically prioritize immediate user satisfaction by passively adapting to current interests [1], which often reinforces the Filter Bubble effect [2], [3] and restricts users from exploring diverse content [4]. To address these limitations, Proactive Recommendation [5], [6] has emerged as a promising research frontier. Instead of randomly presenting diverse content, this paradigm aims to actively enhance user interest toward a specific target item through a sequence of intermediate items, which gradually bridge the gap between the user's initial interests and the target.

This work was supported by the National Natural Science Foundation of China (No. 62576329). *Lei Zhang is the Corresponding author.

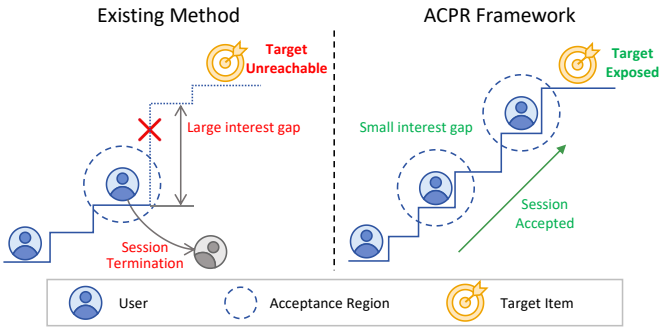


Fig. 1. Illustration of our motivation. (Left) Existing works overlook the user's acceptance limit and may generate items that deviate from the acceptance region, resulting in sequence abandonment. (Right) ACPR explicitly treats user acceptance as a safety constraint, ensuring a feasible path to the target item by confining intermediate steps within the acceptance region.

Existing approaches generally fall into two paradigms. Early efforts rely on supervised learning methods [5], [6], which generate intermediate sequences by navigating latent spaces derived from historical user-item interactions. However, these methods depend heavily on large-scale interaction data that are not originally tailored for proactive objectives, limiting their applicability in proactive scenarios that often involve cold-start issues where interaction signals are sparse. Consequently, recent research has shifted toward Large Language Model (LLM)-driven semantic methods [7], [8]. By leveraging the extensive world knowledge and reasoning ability of LLMs [9], [10], these methods infer connections between items, ensuring the logical coherence of the recommendation sequence.

Despite these advancements, a critical limitation persists: existing methods prioritize maximizing the interest uplift toward the target item. As depicted in Fig 1, these works largely overlook the inherent path dependency of the recommendation sequence, as they assume that users will passively accept all intermediate items. However, according to Expectation Confirmation Theory [11], [12], user acceptance is contingent upon the alignment between recommended content and immediate expectations. Therefore, the above assumption does not always hold and the effective exposure of the target item strictly depends on the continued acceptance of intermediate items, where a rejected step renders the target item unreachable. Consequently, we posit that user acceptance of intermediate items should also be rigorously addressed within the rec-

ommendation process, ensuring that every intermediate item remains within the user's acceptance region to guarantee the successful exposure of the target item.

To realize this insight, we introduce the Acceptance-Constrained Proactive Recommendation (ACPR) framework, which formulates the proactive recommendation as a Constrained Markov Decision Process (CMDP) [13] with user acceptance as a safety constraint. Specifically, we define a computable cost function to quantify the deviation between candidate items and the user's current interests, where recommendations falling outside the acceptance region will incur a cost penalty for safety violations. Unlike standard CMDPs that optimize cumulative rewards, we employ a step-wise greedy strategy to maintain every intermediate item within the user's acceptance region, thereby securing a feasible sequence for effective target exposure. We instantiate the policy using a Large Language Model (LLM) to generate coherent semantic transitions toward the target. To efficiently optimize policy, we employ Reward Constrained Policy Optimization (RCPO) [14] to dynamically balance the proactive objective of interest uplift with safety constraints. Furthermore, we integrate Group Relative Policy Optimization (GRPO) [15] to enhance training stability, leveraging group-based relative feedback to guide policy updates. The contributions are summarized as follows:

- We introduce the Acceptance-Constrained Proactive Recommendation (ACPR) framework for proactive recommendation. To the best of our knowledge, ACPR is the first work to formulate user acceptance as an explicit safety constraint in proactive recommendation.
- We formulate proactive recommendation as a Constrained Markov Decision Process (CMDP) and adopt a step-wise greedy strategy to maintain intermediate items within the user's acceptance region. Based on this formulation, we implement a coarse-to-fine LLM policy and optimize it with RCPO and GRPO.
- Extensive experiments demonstrate that ACPR excels in bridging interest gaps by effectively enhancing users' interest in the target item, which is particularly evident in sparse data scenarios where limited user history hinders a deep understanding of user preferences.

II. Related Work

Unlike traditional Recommender Systems [16]–[18], Proactive Recommendation [5], [6] shifts the paradigm from passively satisfying users' current interests to actively cultivating them toward specific target items. Existing research in this field can be categorized into two types: supervised learning methods and LLM-driven semantic methods. Supervised learning methods generate sequences of intermediate items by navigating latent embedding spaces derived from historical user-item interactions. As a representative work, IRS [5] utilizes a Transformer-based architecture and a self-supervised paradigm to capture interest evolution patterns, decoding a sequence of items that bridges the gap between the user's current interest and the target. IPG [6] introduces a model-agnostic post-processing framework that prioritizes items designed to shift the user's latent state toward the target representation. However, obtaining high-quality embeddings requires large-scale user-item interaction data, making these methods vulnerable to the cold-start scenarios, which are prevalent challenges in proactive recommendation scenarios.

To mitigate the reliance on extensive interaction data, LLM-driven semantic methods leverage the extensive world knowledge and reasoning capabilities of LLM to construct a recommendation sequence through semantic correlations. LLM-IPP [8] employs prompting strategies including Chain-of-Thought (CoT) and Tree-of-Thought (ToT) to infer logical connections between items, generating semantically smooth and explainable sequences. Further advancing this direction, T-PRA [7] introduces a tunable Actor-Critic agent framework to optimize dynamic multi-round interactions. It incorporates user acceptance into a composite reward together with path coherence and distance to the target. Under this soft reward design, a decrease in user acceptance can still be compensated by gains in other reward components. In our work, user acceptance is modeled as an explicit safety constraint, and its penalty strength is adaptively adjusted during optimization.

III. Methodology

The overview of ACPR is illustrated in Fig. 2. We first formulate the task via CMDP in III-A and derive the optimization objective in III-B. Then we present the design of the reward and cost function in III-C, the policy model in III-D, and the optimization algorithm in III-E.

A. Problem Formation with CMDP

Proactive recommendation naturally entails a dual objective where the system should enhance user interest toward a target item while simultaneously maintaining immediate user satisfaction. To capture this trade-off, we formulate this task as a Constrained Markov Decision Process (CMDP) [13], which enables us to optimize the interest uplift under acceptance-related constraints.

A CMDP is defined by a tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{C} \rangle$, where $\mathcal{S}$ denotes the state space reflecting user interest, $\mathcal{A}$ denotes the action space comprising all possible items, $\mathcal{R}$ denotes the reward function, and $\mathcal{C}$ denotes the cost function. For a user at time step t, a policy model π samples an action $a_t \in \mathcal{A}$ based on $s_t \in \mathcal{S}$, receives a reward $r_t = r(s_t, a_t)$ and incurs a cost $c_t = c(s_t, a_t)$ from the environment. This process generates a trajectory of length h:

$$\tau = \{(s_1, a_1, r_1, c_1), \cdots, (s_h, a_h, r_h, c_h)\}. \tag{1}$$

The optimization objective is to maximize the cumulative reward subject to an average cost threshold α:

$$\begin{aligned} \max_{\pi} \quad & \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=1}^{h} r_t\right], \\ \text{s.t.} \quad & \mathbb{E}_{\tau\sim\pi}\left[\frac{1}{h}\sum_{t=1}^{h} c_t\right] \le \alpha. \end{aligned} \tag{2}$$

This objective encourages the policy to improve user interest toward the target item while limiting the average deviation from the user's current interests along the recommendation trajectory. The threshold α can be interpreted as the tolerated average deviation from the user's current interests.

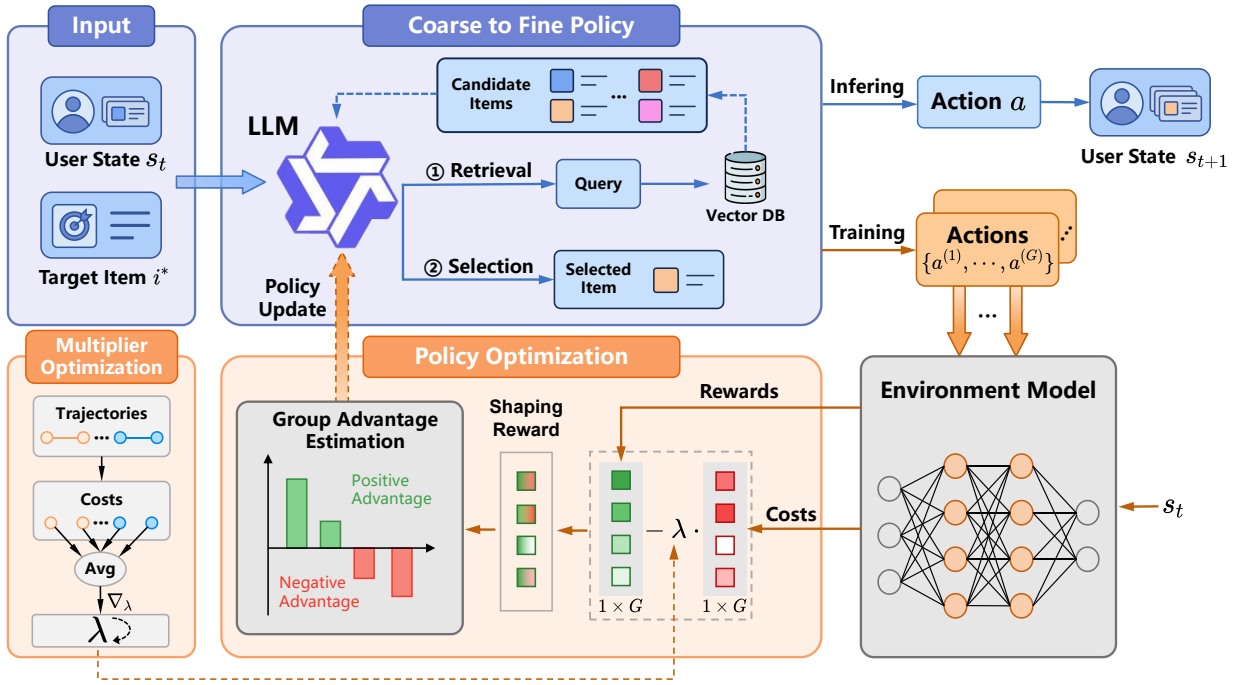


Fig. 2. Overview of the Acceptance-Constrained Proactive Recommendation (ACPR) framework. Conditioned on the user state s_t and target item i^*, the Coarse-to-Fine Policy derives an action a via sequential retrieval and selection stages. During inference, the policy interacts iteratively to evolve the user state until the target is accepted or the session is terminated. During training, ACPR employs step-wise optimization where a group of candidate actions $\{a^{(1)}, \ldots, a^{(G)}\}$ is sampled for the current state. An Environment Model yields both immediate rewards and costs, which are synthesized into shaping rewards. These rewards are processed by GRPO to estimate the relative advantages of actions, ultimately generating gradients to optimize the policy. After multiple policy updates, ACPR samples trajectories from the latest policy to estimate the average cost and adjust λ, ensuring the penalty is adapted to the latest policy.

B. Step-wise Optimization Objective

Although Eq. 2 provides a natural trajectory-level formulation, it does not fully reflect the failure mode of proactive recommendation. In practice, a single intermediate item that deviates significantly from the user's current interests may interrupt the interaction, making the remaining trajectory rewards unattainable. Therefore, the acceptance risk should be controlled at every step, so that each intermediate item remains within the user's current acceptance region. To motivate a step-wise alternative, we note that

$$\max_{\pi} \mathbb{E}\left[\sum_{t=1}^{h} r_t\right] \leq \sum_{t=1}^{h} \max_{\pi} \mathbb{E}\left[r_t\right]. \tag{3}$$

This upper bound suggests that improving the immediate reward at each step can serve as a reasonable proxy for improving the overall trajectory reward. Following this intuition, we adopt the following step-wise surrogate objective:

$$\begin{aligned} &\max_{\pi} J_{\pi}^{R} = \mathbb{E}_{a_t \sim \pi(\cdot|s_t)}[r_t], \\ &\text{s.t. } J_{\pi}^{C} = \mathbb{E}_{a_t \sim \pi(\cdot|s_t)}[c_t] \leq \alpha. \end{aligned} \tag{4}$$

Here, J_{π}^{R} and J_{π}^{C} denote the step-wise expected reward and cost, respectively. This formulation aligns with the goal of maintaining continuous user engagement: at each step t, the policy model selects an action that improves the immediate potential gain toward the target i^* while keeping the immediate acceptance cost under the threshold.

C. Design of Reward and Cost Function

In sequential recommendation scenarios, we represent the user state as the interaction history sequence. The initial state is the observed interaction history, $s_1 = (i_1, i_2, \cdots, i_n)$, where each i_k denotes an interacted item. In the offline setting, after the policy recommends an item a_t, we construct the next state by appending it to the history:

$$s_{t+1} = (i_1, i_2, \cdots, i_n, a_1, \cdots, a_t) \tag{5}$$

This deterministic update is an offline approximation because real-time user feedback is unavailable during training. The rollout terminates once the target item i^* is recommended.

For a specific user and corresponding state s_t, we define a potential function $\Phi(s_t) = \Pr(i^*|s_t)$ as the predicted interaction probability of the target item i^*. The primary objective of proactive recommendation is to maximize the increase of interest toward the target from the initial state to the end of the trajectory, denoted as $\Pr(i^*|s_h) - \Pr(i^*|s_1) = \Phi(s_h) - \Phi(s_1)$. By applying the telescoping sum expansion, this trajectory-level gain can be decomposed into step-wise increments:

$$\Phi(s_h) - \Phi(s_1) = \sum_{t=1}^{h-1} \left[\Phi(s_{t+1}) - \Phi(s_t)\right]. \tag{6}$$

Each term in this summation quantifies the immediate progress toward the target item driven by the action a_t. Consequently, the immediate reward function can be naturally designed as the potential difference: $r_t = \Phi(s_{t+1}) - \Phi(s_t)$. To encourage the model to seize the appropriate opportunity for recommending

the target item, we assign the absolute interaction probability $\Pr(i^*|s_t)$ as the terminal reward. The final shaped reward function is concluded as:

$$r_t = \begin{cases} \Phi(s_{t+1}) - \Phi(s_t), & a_t \neq i^*, \\ \Phi(s_t), & a_t = i^*. \end{cases} \tag{7}$$

To incorporate user acceptance into the CMDP, we define the cost function based on interest deviation. We posit that an item is accepted only if its interaction probability exceeds a confidence threshold δ; probabilities below this threshold are deemed unsafe, incurring a cost. Accordingly, we employ a normalized cost motivated by hinge loss:

$$c_t = \max\left(0, 1 - \Pr(a_t|s_t)/\delta\right), \tag{8}$$

where a larger gap between $\Pr(a_t|s_t)$ and δ results in a higher cost, thereby guiding the policy to avoid risky exploration that might harm user experience.

The implementation of the above reward and cost functions relies on the estimation of the interaction probability $\Pr(a|s)$. Given the offline nature of our training and the absence of real-time user feedback, we approximate this probability using a standard click-through rate prediction model, DIN [19], as the environment model. We then use DIN to estimate the interaction probability for any state-action pair:

$$\Pr(a|s) \approx \text{DIN}(a; s). \tag{9}$$

DIN serves as an offline approximation of user feedback, providing the interaction estimates required for reward and cost computation.

D. Policy Model

We employ a Large Language Model (LLM) as the policy model to generate intermediate items by leveraging its world knowledge and semantic reasoning capabilities. However, directly evaluating all items is computationally infeasible under a limited context window. Instead, we design a hybrid coarse-to-fine architecture with two stages: Generative Retrieval and Discriminative Selection.

1) Generative Retrieval: This stage narrows the massive search space to a manageable candidate set. At step t, guided by the retrieval instruction I_R, the LLM generates an ideal next-item description $d_t = \text{LLM}(s_t, i^*|I_R)$ (e.g., specific genres or features) for the proactive objective. This description is embedded into a query vector $e_t = \text{Embed}(d_t)$ to search the vector database, and then retrieve the top-k items that are semantically closest to this ideal description:

$$C_t = \text{Top}_k\left\{\cos(e_t, e_i) \mid i \in \mathcal{I}\right\}. \tag{10}$$

2) Discriminative Selection: This stage acts as a reranker to pick the optimal action from the retrieved candidates C_t. The LLM evaluates the specific content of items in C_t to satisfy the objective, outputting the final recommendation:

$$a_t = \text{LLM}(s_t, i^*, C_t \mid I_S). \tag{11}$$

This two-stage design effectively bridges the gap between the massive action space of recommendation systems and the context constraints of LLMs.

Algorithm 1. Primal-Dual Optimization

Require: User set $\mathcal{U}$, Group size G, Click threshold δ, Tolerance factor α, Learning rates η_λ.

Initialize: Policy parameters θ, multiplier $\lambda \leftarrow \lambda_0$.
for $k = 0, 1, \ldots$ **do**
  Sample users $\mathcal{U}_k \subset \mathcal{U}$, generate trajectories $\mathcal{D}_k$ via π_θ
  for $s \in \mathcal{D}_k$ **do**
    Sample G actions $\{a^{(1)}, \ldots, a^{(G)}\} \sim \pi_{\theta_{\text{old}}}(\cdot|s)$
    Calculate shaping rewards: $\hat{\mathbf{r}}$ via Eq.(13)
    Compute advantages $A^{(g)}$ via Eq. (15) and update policy π_θ to maximize GRPO objective Eq. (16)
  end for
  Estimate average cost $\hat{J}_\pi^C$ and update Lagrangian multiplier: $\lambda_{k+1} \leftarrow \Gamma_\lambda[\lambda_k + \eta_\lambda(k)(\hat{J}_{\pi_\theta}^C - \alpha)]$
end for

E. Optimization with RCPO and GRPO

CMDPs are typically solved via the Lagrange relaxation technique [20], which transforms the constrained optimization problem into an unconstrained saddle-point optimization problem. We relax the constraint defined in Eq. 4 by introducing a Lagrangian multiplier $\lambda \geq 0$. This multiplier acts as a regulator that balances the trade-off between the original reward maximization and the penalty for constraint violation:

$$\min_{\lambda \geq 0} \max_{\theta} \mathcal{L}(\lambda, \theta) = \min_{\lambda \geq 0} \max_{\theta} \Big[J_{\pi_\theta}^R - \lambda (J_{\pi_\theta}^C - \alpha) \Big]. \tag{12}$$

We adopt the Reward Constrained Policy Optimization (RCPO) [14] framework to solve this min-max objective via a two-timescale stochastic approximation. RCPO updates the policy on a faster timescale to optimize the Lagrangian quasi-statically, while the multiplier is updated on a slower timescale to enforce constraints gradually.

To implement the two-timescale mechanism efficiently, we employ a primal-dual alternating update scheme. At each training iteration k, we first execute the policy update stage (primal). We fix the multiplier λ_k and update the policy π_θ to maximize the expected Lagrangian shaping reward:

$$\hat{r}_t(\lambda) = r_t - \lambda c_t. \tag{13}$$

After multiple policy updates, we switch to multiplier update stage (dual). We sample trajectories from the latest policy to estimate the average cost $\hat{J}_{\pi_\theta}^C$ and then adjust λ via projected gradient ascent:

$$\lambda_{k+1} = \Gamma_\lambda \Big[\lambda_k + \eta_\lambda (\hat{J}_{\pi_\theta}^C - \alpha) \Big], \tag{14}$$

where Γ_λ projects λ into the range $[0, \lambda_{\max}]$. This alternating schedule ensures that the policy is optimized under the current penalty level before λ is updated, thereby effectively meeting the two-timescale requirement.

To optimize the policy model without training a separate value model, we adopt Group Relative Policy Optimization (GRPO) [15]. Specifically, for a given state s, we sample a group of G candidate actions $\{a^{(1)}, \ldots, a^{(G)}\}$ from the policy $\pi_{\theta_{\text{old}}}$ and compute corresponding Lagrangian shaping rewards

TABLE I
STATISTICS OF DATASETS AFTER PREPROCESSING

Dataset	#Users	#Items	Avg. Inter. Per User	Avg. Inter. Per Item	Density
Steam	18,422	6,649	13.28	36.80	0.20%
MovieLens-1M	6,034	3,533	95.33	162.82	2.70%

Note: Avg. Inter. stands for Average Interaction.

$\hat{\mathbf{r}} = \{\hat{r}^{(1)}, \ldots, \hat{r}^{(G)}\}$. GRPO estimates the advantage $A^{(g)}$ for each action via group-relative normalization:

$$A^{(g)} = \frac{\hat{r}^{(g)} - \text{mean}(\hat{\mathbf{r}})}{\text{std}(\hat{\mathbf{r}})}. \tag{15}$$

GRPO integrates the clipped surrogate loss with a KL-divergence penalty to ensure stable updates for policy:

$$J(\theta) = \frac{1}{G}\sum_{g=1}^{G} L^{(g)}(\theta) - \beta\, \mathbb{D}_{\text{KL}}(\pi_\theta \,\|\, \pi_{\text{ref}})\,. \tag{16}$$

Here, $L^{(g)}(\theta)$ is the clipped objective for the g-th sample:

$$L^{(g)}(\theta) = \min\Big(\rho^{(g)} A^{(g)},\ \text{clip}(\rho^{(g)}, 1-\epsilon, 1+\epsilon) A^{(g)}\Big)\,, \tag{17}$$

where ϵ is the clipping hyperparameter, and $\rho^{(g)}$ represents the probability ratio between the current and old policy:

$$\rho^{(g)} = \frac{\pi_\theta(a^{(g)} \mid s)}{\pi_{\theta_{\text{old}}}(a^{(g)} \mid s)}. \tag{18}$$

This objective guides the policy toward high rewards while preventing excessive deviation from the reference model.

Algorithm 1 summarizes the optimization procedure.

IV. EXPERIMENT

A. Experimental Setup

1) Datasets: We evaluate our method on two real-world datasets: Steam [21] and MovieLens-1M [22] datasets. In the preprocessing, we first sort all user interactions chronologically, remove users with fewer than 5 interactions, and exclude items without interactions. Then, for Steam dataset, interactions with "play hours" greater than 3.0 are treated as positive feedback; for MovieLens-1M dataset, ratings greater than 3 are retained as positive feedback. The resulting statistics are shown in Table I. With an information density an order of magnitude lower than MovieLens, the Steam dataset reflects real-world sparse and cold-start scenarios, thereby better validating the practical effectiveness of the proposed method. For the proactive setting, we randomly sample a non-interacted item as the target item i^*.

2) Evaluation Metrics: Following prior studies [5], [8], we adopt four established metrics. First, Increment of Interest (IoI) and Increment of Rank (IoR) measure the uplift in user interest toward the target item. IoI is defined as the difference in the logarithmic interaction probability of the target item between the final state and the initial state, while IoR is defined as the difference in the target item's rank between these two states. Second, Log-Perplexity (Log-PPL) and Acceptability (Acp) evaluate overall sequence satisfaction. Log-PPL is defined as the negative average logarithmic interaction probability across the sequence, while Acp is the average rank of all items in the sequence.

Furthermore, we propose Session Survival Rate (SSR) to measure interaction reliability. A session is counted as compliant only if every item satisfies the top-m rank constraint. Formally, SSR is the proportion of fully compliant sessions:

$$\text{SSR} = \frac{1}{|\mathcal{D}_{\text{test}}|}\sum_{\tau \in \mathcal{D}_{\text{test}}} \mathbb{I}\left(\max_{(s,a)\in\tau} \text{Rank}(a|s) \le m\right), \tag{19}$$

where $\mathbb{I}(\cdot)$ is the indicator function. Note that m serves as a tolerance threshold for user dissatisfaction, where the total number of candidates refers to the candidate set size at each step. To avoid an overly strict criterion under which nearly all methods would fail, we set m to 50% of the candidate set size in all experiments.

As these metrics rely on the interaction probability $\Pr(a|s)$, following previous works, we adopt an independent SASRec [23] model as the offline evaluator. This provides a consistent offline approximation of user feedback, although it does not replace online evaluation with real users.

3) Baselines: The proposed ACPR is compared with the following representative baselines. IRS [5] learns transition patterns in the latent embedding space and uses a Transformer-based decoder to generate intermediate items. LLM-IPP [8] formulates the recommendation process as a logical reasoning task and leverages LLM prompting techniques to construct semantically coherent paths. T-PRA [7] models the task as a sequential decision-making problem, adopting an Actor-Critic architecture and incorporating Direct Preference Optimization (DPO) to refine the policy based on interactive feedback.

B. Implementation Details

We initialize the policy model π_θ with Qwen3-4B-Instruct [24] and fine-tune it via LoRA [25] with a rank of $r = 64$. In the generative retrieval stage, we employ Qwen3-Embedding-0.6B [26] to encode both item metadata and queries, and set the retrieval size to $k = 40$. We train DIN and SASRec using the RecBole framework [27]. GRPO is implemented with VeRL [28] using a group size of $G = 4$. We optimize the policy with a learning rate of 1×10^{-5} for 20 iterations, where each iteration samples a batch of users of size $|D_k| = 120$. The multiplier is initialized as $\lambda_0 = 0.1$ and updated with learning rate $\eta_\lambda = 0.1$, and is clipped the multiplier within $[0, \lambda_{\max}]$ with $\lambda_{\max} = 1$. We empirically set the acceptance threshold to $\delta = 0.05$ and the average cost threshold $\alpha = 0.05$. All experiments are conducted on 3 NVIDIA L40S GPUs.

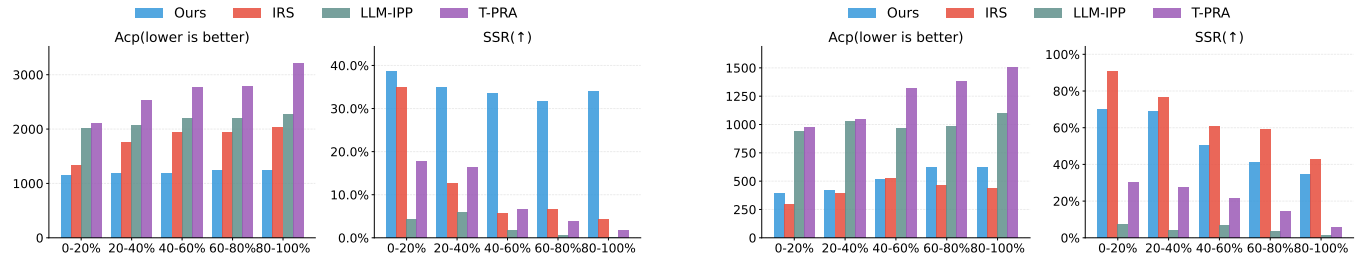
C. Comparison with SOTA Methods

Table II reports the results on Steam and MovieLens-1M. Overall, ACPR achieves the best acceptance-related performance on the sparse Steam dataset and remains competitive on the dense MovieLens-1M dataset, showing a favorable balance between target guidance and interaction feasibility.

Specifically, on the Steam dataset, ACPR achieves the best SSR (33.68%) and the lowest Acp (1206.22), substantially

TABLE II
PERFORMANCE COMPARISON WITH SOTA METHODS ON STEAM AND MOVIELENS-1M. ACPR ACHIEVES THE BEST ACCEPTANCE-RELATED PERFORMANCE ON THE SPARSE STEAM DATASET AND REMAINS COMPETITIVE ON THE DENSE MOVIELENS-1M DATASET.

Method	Steam					MovieLens-1M				
	SSR(↑)	IoI(↑)	IoR(↑)	Log-PPL(↓)	Acp(↓)	SSR(↑)	IoI(↑)	IoR(↑)	Log-PPL(↓)	Acp(↓)
IRS [5]	16.55%	0.351	-37.92	9.09	1904.67	64.28%	2.88	713.82	7.15	353.50
LLM-IPP [8]	2.28%	0.330	14.96	9.04	2184.40	4.36%	3.89	558.69	9.13	1006.76
T-PRA [7]	10.83%	0.288	50.76	9.79	2660.31	19.42%	3.30	532.48	10.14	1261.91
Ours	33.68%	0.454	-0.82	8.68	1206.22	52.44%	2.94	626.44	7.87	506.21



(a) Performance on Steam (Sparse)

(b) Performance on MovieLens-1M (Dense)

Fig. 3. **Robustness analysis across varying target difficulties.** The x-axis represents the normalized ranking percentile in the initial user preference list of the target item (Top-N%), where a higher percentage indicates a greater gap between the user's initial interest and the target. We report Acp (↓) and SSR (↑) on (a) the sparse Steam dataset and (b) the dense MovieLens-1M dataset. Note that our method maintains a highly stable Acp and a resilient SSR, particularly in the hardest intervals (80-100%) on Steam.

outperforming the strongest baseline IRS (16.55%). On the MovieLens-1M dataset, IRS remains the strongest method on SSR and Acp. We attribute this result to the varying sensitivity of models to data sparsity. MovieLens-1M is relatively dense (2.70% density), allowing IRS, which is based on supervised learning, to capture explicit statistical correlations between users and items effectively. However, this advantage is dependent on data density. As evidenced by the performance drop on the Steam dataset (0.2%), supervised learning methods face limitations in sparse scenarios. Conversely, our method leverages semantic reasoning to maintain robust performance consistency across datasets with varying sparsity, making it more adaptable to complex recommendation environments where interactions are often scarce. These findings further underscore the practical applicability of our proposed approach in diverse real-world settings.

Further analysis of the metrics reveals a distinct Guidance-Compliance Trade-off in proactive recommendation tasks. Specifically, we observe that baselines achieving higher IoI or IoR often suffer from degraded performance in Acp and Log-PPL, alongside lower SSR. This suggests that while aggressive recommendation strategies can theoretically induce significant shifts towards the target item, the resulting drastic deviation from current interest often compromises interaction fluency, thereby increasing the risk of session abandonment. In this context, ACPR demonstrates a more favorable balance within this trade-off. By incorporating explicit satisfaction constraints within the CMDP framework, we confine policy optimization to a range that remains acceptable to the user. The empirical results indicate that, while our method may not always yield the highest magnitude of interest increase (e.g., in IoI) compared to the most aggressive baselines, it achieves a substantial advantage in maintaining high SSR and low Acp. These factors are vital for maintaining a functional recommendation sequence in real-world scenarios, which is essential to successfully exposing the target item. These results demonstrate that ACPR effectively balances "effective guidance" and "user retention", achieving sustainable guidance without alienating the user.

D. Robustness across Target Difficulties

In practice, target items are often far from a user's current interest, making them initially hard to recommend. Since aggregate metrics can obscure performance on such challenging cases, we conduct a fine-grained analysis based on target difficulty to assess ACPR's robustness in bridging these interest gaps. Target Difficulty is quantified using the target item's normalized ranking percentile in the initial user preference list. We partition test samples into five 20% intervals and visualize performance in Fig. 3, where a higher percentile indicates a larger interest gap between the target item and the user.

On the sparse Steam dataset, ACPR demonstrates an advantage over all baselines. It maintains robustness across all difficulty levels with relatively small performance fluctuation. Specifically, ACPR achieves the best Acp across all difficulty levels and maintains the most stable SSR. Notably, in high-difficulty scenarios where the interest gap is large (ranked after 50%), ACPR delivers a recommendation pipeline superior to the baselines. By contrast, the other methods exhibit higher Acp and a sharper decline in SSR as the target difficulty increases. This result suggests that the explicit acceptance

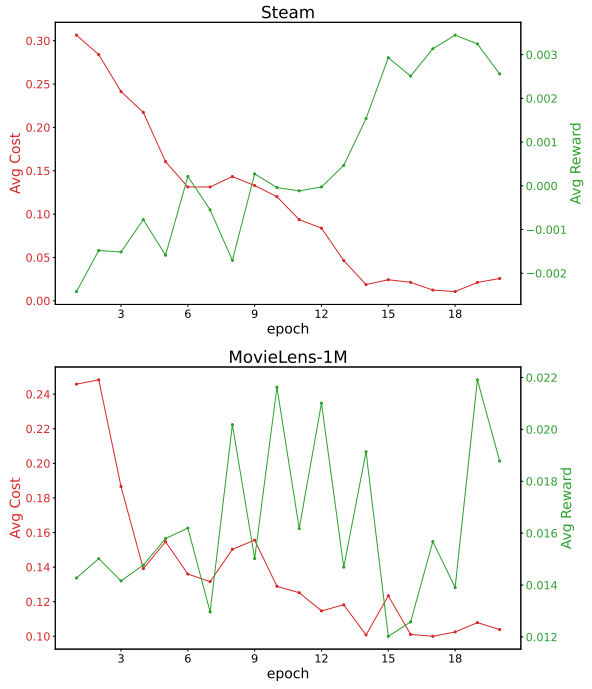


Fig. 4. **Training convergence curves on MovieLens-1M and Steam datasets.** The red line (left axis) represents the average step-wise cost, and the green line (right axis) denotes the average reward.

constraint in ACPR helps maintain a feasible recommendation path even in sparse and difficult scenarios.

Meanwhile, on the dense MovieLens-1M dataset, the performance gap between ACPR and IRS narrows considerably. Here, IRS remains the strongest method overall, while our method achieves results comparable to IRS and has a more stable acceptance performance than the other LLM-based baselines as difficulty increases. This suggests that acceptance constraints help maintain interaction feasibility in medium and high difficulty cases.

E. Training Convergence Analysis

Figure 4 shows the optimization dynamics of ACPR on the two datasets, illustrating the efficacy of the proposed constrained optimization.

On the Steam dataset, the curves demonstrate the framework's efficiency in adapting the generative policy to the dual objectives. The policy initially exhibits high constraint violations, but the average cost drops below the tolerance threshold α around Epoch 13, accompanied by a steady rise in reward. This suggests that ACPR does not merely suppress unsafe behaviors but successfully learns valid guidance trajectories within the feasible region defined by the user's interests.

On MovieLens-1M, the cost initially decreases and then oscillates around 0.1, suggesting a more constrained feasible region in the denser interaction space. Meanwhile, the average reward remains positive and trends upward, indicating that the policy continues to explore valid guidance opportunities instead of collapsing into a trivial conservative strategy (i.e., merely recommending safe items to minimize cost).

TABLE III
ABLATION RESULTS ON THE MOVIELENS-1M DATASET

Method	**SSR**(↑)	**IoI**(↑)	**IoR**(↑)	**Log-PPL**(↓)	**Acp**(↓)
LLM-IPP	4.36%	3.89	558.69	9.13	1006.76
ACPR w/o train	6.85%	4.05	756.80	9.12	968.02
ACPR w/o constrain	16.01%	3.36	644.82	9.30	919.34
ACPR	52.44%	2.94	626.44	7.87	506.21

F. Ablation Study

We study the contribution of each component of ACPR on the MovieLens-1M dataset under two settings: without training (ACPR w/o train) and without the Lagrangian constraint (ACPR w/o constrain). As shown in Table III, even without training, our coarse-to-fine architecture outperforms the prompt-based LLM-IPP in SSR (6.85% vs. 4.36%) and IoR (756.80 vs. 558.69). This demonstrates the effectiveness of our architectural design in narrowing the action space and filtering irrelevant candidates. Furthermore, when the constraint term is removed, the average rank still remains at a high value of 919.34, suggesting that the model tends to adopt aggressive recommendation strategies. In contrast, the complete ACPR achieves a substantial improvement in SSR to 52.44% while significantly improving the average rank to 506.21. This result demonstrates that the Lagrangian constraint successfully imposes a safety boundary, enabling the policy model to maximize user interest uplift toward the target item while effectively satisfying current user interests.

G. Case Study

We conduct a case study comparing ACPR with the best baseline IRS, as shown in Table IV. In this case, the target item belongs to the Comedy and Horror genres, while the user's recent history is dominated by Children's and Animation content, with no prior interaction with Horror. ACPR better bridges this interest gap by starting from items that overlap with the user's history and the target's genre, and further exploiting the series relation before recommending the target item. In contrast, IRS captures the broad Comedy correlation but introduces irrelevant genres such as War and Action, and struggles to narrow down the recommendation scope from broad genres to the specific category. It also lacks the ability to identify the optimal timing to introduce the target item, resulting in a continuously prolonged sequence. These observations highlight ACPR's advantage in planning semantic transitions toward the target item.

V. CONCLUSION

In this paper, we introduce a novel Acceptance-Constrained Proactive Recommendation framework (ACPR) for proactive recommendation. To the best of our knowledge, ACPR is the first work to formulate user acceptance as an explicit safety constraint for effective target exposure. ACPR formulates Proactive Recommendation as a CMDP and introduces a step-wise greedy strategy to ensure each recommended item stays within the user's acceptance region. We implement an LLM-based policy model and optimize it with RCPO and GRPO.

TABLE IV
CASE STUDY COMPARISON BETWEEN ACPR AND IRS ON MOVIELENS-1M DATASET.

User History (Genre Counts in Last 20 Interactions): Children's (10), Animation (9), Sci-Fi (7), Musical (6), Thriller (5), ... , Comedy (2), Horror (0)
Target: The Toxic Avenger Part III: The Last Temptation of Toxie; **Genre**: Comedy, Horror; **Initial Rank: 2352**

Ours (ACPR)			Baseline (IRS)		
Movie Name	**Genre**	**Rank**	**Movie Name**	**Genre**	**Rank**
Honey, I Blew Up the Kid	Children's, Comedy, Sci-Fi	38	Shanghai Noon	Action	924
Beetlejuice	Comedy, Fantasy	531	But I'm a Cheerleader	Comedy	149
Austin Powers: International Man of Mystery	Comedy	310	Mars Attacks!	Action, Comedy, Sci-Fi, War	115
Austin Powers: The Spy Who Shagged Me, The Toxic Avenger	Comedy	5	Tampopo	Comedy	566
The Toxic Avenger, Part II	Comedy, Horror	1262	Airplane!	Animation, Children's, Comedy	73
Bad Taste	Comedy, Horror	746	Chicken Run	Action, Adventure, Comedy	2
The Toxic Avenger Part III: The Last Temptation of Toxie	Comedy, Horror	333	Duck Soup	Comedy, War	531
(Finished)			(More Items ...)		
			The Toxic Avenger Part III: The Last Temptation of Toxie	Comedy, Horror	558

Extensive experiments demonstrate that ACPR can effectively uplift user interest toward target items while significantly improving the feasibility of recommendation sequences compared to existing baselines.

REFERENCES

[1] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM computing surveys (CSUR)*, vol. 52, no. 1, pp. 1–38, 2019.

[2] E. Pariser, *The filter bubble: What the Internet is hiding from you*. penguin UK, 2011.

[3] C. Gao, S. Wang, S. Li, J. Chen, X. He, W. Lei, B. Li, Y. Zhang, and P. Jiang, "Cirs: Bursting filter bubbles by counterfactual interactive recommender system," *ACM Transactions on Information Systems*, vol. 42, no. 1, pp. 1–27, 2023.

[4] T. T. Nguyen, P.-M. Hui, F. M. Harper, L. Terveen, and J. A. Konstan, "Exploring the filter bubble: the effect of using recommender systems on content diversity," in *Proceedings of the 23rd international conference on World wide web*, 2014, pp. 677–686.

[5] H. Zhu, H. Ge, X. Gu, P. Zhao, and D. L. Lee, "Influential recommender system," in *2023 IEEE 39th International Conference on Data Engineering (ICDE)*. IEEE, 2023, pp. 1406–1419.

[6] S. Bi, W. Wang, H. Pan, F. Feng, and X. He, "Proactive recommendation with iterative preference guidance," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 871–874.

[7] M. Wang, C. Gao, W. Wang, Y. Li, and F. Feng, "Tunable llm-based proactive recommendation agent," in *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025, pp. 19 262–19 276.

[8] M. Wang, S. Bi, W. Wang, C. Gao, Y. Li, and F. Feng, "Leveraging llms for influence path planning in proactive recommendation," in *Companion Proceedings of the ACM on Web Conference 2025*, 2025, pp. 1355–1359.

[9] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022.

[10] L. Wu, Z. Zheng, Z. Qiu, H. Wang, H. Gu, T. Shen, C. Qin, C. Zhu, H. Zhu, Q. Liu *et al.*, "A survey on large language models for recommendation," *World Wide Web*, vol. 27, no. 5, p. 60, 2024.

[11] R. L. Oliver, "A cognitive model of the antecedents and consequences of satisfaction decisions," *Journal of Marketing Research*, vol. 17, no. 4, pp. 460–469, 1980.

[12] G. Mandler, "The structure of value: Accounting for the affective effects of schema congruity and incongruity," in *Affect and cognition: The seventeenth annual Carnegie symposium on cognition*, M. S. Clark and S. T. Fiske, Eds. Psychology Press, 1982, pp. 3–36.

[13] E. Altman, *Constrained Markov decision processes*. Routledge, 2021.

[14] C. Tessler, D. J. Mankowitz, and S. Mannor, "Reward constrained policy optimization," in *7th International Conference on Learning Representations*, 2019.

[15] Z. Shao, P. Wang, Q. Zhu, R. Xu, J. Song, X. Bi, H. Zhang, M. Zhang, Y. Li, Y. Wu *et al.*, "Deepseekmath: Pushing the limits of mathematical reasoning in open language models," *arXiv preprint arXiv:2402.03300*, 2024.

[16] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[17] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations," in *Proceedings of the 10th ACM conference on recommender systems*, 2016, pp. 191–198.

[18] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th international conference on world wide web*, 2017, pp. 173–182.

[19] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, "Deep interest network for click-through rate prediction," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 1059–1068.

[20] D. P. Bertsekas, "Nonlinear programming," *Journal of the Operational Research Society*, vol. 48, no. 3, pp. 334–334, 1997.

[21] A. Pathak, K. Gupta, and J. McAuley, "Generating and personalizing bundle recommendations on steam," in *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*, 2017, pp. 1073–1076.

[22] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *Acm transactions on interactive intelligent systems (tiis)*, vol. 5, no. 4, pp. 1–19, 2015.

[23] W.-C. Kang and J. McAuley, "Self-attentive sequential recommendation," in *2018 IEEE international conference on data mining (ICDM)*. IEEE, 2018, pp. 197–206.

[24] Q. Team, "Qwen3 technical report," 2025. [Online]. Available: https://arxiv.org/abs/2505.09388

[25] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, "Lora: Low-rank adaptation of large language models." *ICLR*, vol. 1, no. 2, p. 3, 2022.

[26] Y. Zhang, M. Li, D. Long, X. Zhang, H. Lin, B. Yang, P. Xie, A. Yang, D. Liu, J. Lin, F. Huang, and J. Zhou, "Qwen3 embedding: Advancing text embedding and reranking through foundation models," *arXiv preprint arXiv:2506.05176*, 2025.

[27] W. X. Zhao, S. Mu, Y. Hou, Z. Lin, Y. Chen, X. Pan, K. Li, Y. Lu, H. Wang, C. Tian, Y. Min, Z. Feng, X. Fan, X. Chen, P. Wang, W. Ji, Y. Li, X. Wang, and J. Wen, "Recbole: Towards a unified, comprehensive and efficient framework for recommendation algorithms," in *CIKM*. ACM, 2021, pp. 4653–4664.

[28] G. Sheng, C. Zhang, Z. Ye, X. Wu, W. Zhang, R. Zhang, Y. Peng, H. Lin, and C. Wu, "Hybridflow: A flexible and efficient rlhf framework," in *Proceedings of the Twentieth European Conference on Computer Systems*, 2025, pp. 1279–1297.