G-UBS: Towards Robust Understanding of Implicit Feedback via Group-Aware User Behavior Simulation

Boyu Chen[1,2,3,*], Siran Chen[1,2,3,*], Zhengrong Yue[6,*], Kainan Yan[1,2,*], Chenyun Yu[5,†], Beibei Kong[3], Lei Cheng[3], Chengxiang Zhuo[3], Zang Li[3], Yali Wang[1,4,†]

[1] Shenzhen Key Laboratory of Computer Vision and Pattern Recognition, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China
[2] University of Chinese Academy of Science, Beijing, China
[3] Platform and Content Group, Tencent, Shenzhen, China
[4] Shanghai Artificial Intelligence Laboratory, Shanghai, China
[5] Shenzhen Campus of Sun Yat-sen University, Shenzhen, China
[6] Shanghai Jiaotong University, Shanghai, China
chenboyu18, chensiran17@mails.ucas.ac.cn, yuchy35@mail.sysu.edu.cn, yl.wang@siat.ac.cn

Abstract

User feedback is critical for refining recommendation systems, yet explicit feedback (e.g., likes or dislikes) remains scarce in practice. As a more feasible alternative, inferring user preferences from massive implicit feedback has shown great potential (e.g., a user quickly skipping a recommended video usually indicates disinterest). Unfortunately, implicit feedback is often noisy: a user might skip a video due to accidental clicks or other reasons, rather than disliking it. Such noise can easily misjudge user interests, thereby undermining recommendation performance. To address this issue, we propose a novel Group-aware User Behavior Simulation (G-UBS) paradigm, which leverages contextual guidance from relevant user groups, enabling robust and in-depth interpretation of implicit feedback for individual users. Specifically, G-UBS operates via two key agents. First, the User Group Manager (UGM) effectively clusters users to generate group profiles utilizing a "summarize-cluster-reflect" workflow based on LLMs. Second, the User Feedback Modeler (UFM) employs an innovative group-aware reinforcement learning approach, where each user is guided by the associated group profiles during the reinforcement learning process, allowing UFM to robustly and deeply examine the reasons behind implicit feedback. To assess our G-UBS paradigm, we have constructed a Video Recommendation benchmark with Implicit Feedback (IF-VR). To the best of our knowledge, this is the first multi-modal benchmark for implicit feedback evaluation in video recommendation, encompassing 15k users, 25k videos, and 933k interaction records with implicit feedback. Extensive experiments on IF-VR demonstrate that G-UBS significantly outperforms mainstream LLMs and MLLMs, with a 4.0% higher proportion of videos achieving a play rate $>$30% and 14.9% higher reasoning accuracy on IF-VR.

Introduction

Nowadays, multi-modal content platforms like TikTok, Kuaishou, and Tencent Video have become integral to our daily lives. To provide effective and personalized services, these platforms strive to establish accurate recommendation systems by learning user preferences from interaction records, attribute information, and user feedback. However, when using these platforms, users are rarely active in providing explicit feedback (Xie et al. 2021) (e.g., ratings, likes, dislikes, and their underlying reasons). Instead, indirect behavioral cues are more observable, such as quick video skips, non-clicks, and low completion rates, which serve as implicit feedback reflecting user discontent. Consequently, in-depth interpretation of implicit feedback is crucial for boosting recommendation accuracy and personalization. However, implicit feedback typically contains substantial noise, which can easily lead to misjudgment of user interests, thereby impairing recommendation performance and ultimately leading to user attrition and platform abandonment (Zhao et al. 2023). For instance, a quick skip may result from accidental operations (e.g., one-handed usage), user habits, or environmental interference, rather than genuine disinterest in the content. This presents a key challenge: *how to robustly discern the underlying causes of users' implicit feedback in the presence of noisy signals?*

Current efforts to mine implicit feedback can be categorized into embedding-based and LLM-based approaches. Traditional embedding-based approaches (Chen et al. 2021; He et al. 2016; Park and Lee 2022; Guo et al. 2017) map all implicit feedback into embedding features and feed them into recommendation models under the assumption that richer features will produce better performance. Since these schemes cannot truly capture why users dislike certain content, they tend to result in poor interpretability. LLM-based approaches leverage large language models (LLMs) (Zhang et al. 2025, 2024c; Yang et al. 2024b) or RL-tuned models (Yang et al. 2025b; Zhao et al. 2023, 2018) to understand and simulate user behavior (e.g., predicting whether a user will like an item). However, existing LLM-based methods focus predominantly on text modality, lacking the ability to jointly perceive information across multiple modalities. In addition, they fail to address the noise in the individual's implicit feedback, further limiting their performance.

[*] Equal contribution.
[†] Corresponding author.

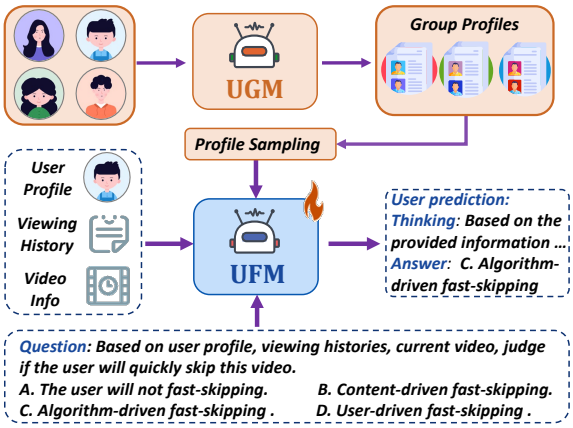

Figure 1: Overview of our G-UBS paradigm: To better visualize implicit feedback, we integrate UGM-generated group profiles into the UFM training process.

To address these challenges, we propose a novel Group-Aware User Behavior Simulation (**G-UBS**) paradigm, aiming to robustly and profoundly understand users' implicit feedback under the contextual guidance of relevant user groups as shown in Fig 1. Specifically, G-UBS comprises two key collaborative agents, namely User Group Manager (**UGM**) and User Feedback Modeler (**UFM**). The UGM agent is designed to support 1,000 concurrent users and generate up to 50 distinct group profiles via an LLM-powered "summarize-cluster-reflect" workflow. The UFM agent integrates group profiles from UGM and multi-modal information to optimize the training of individual user simulators, effectively filtering noise in implicit negative feedback. In summary, G-UBS is a pioneering paradigm that ensures robust scalability while enabling more accurate and reliable simulations to better understand implicit user feedback.

To evaluate our proposed method, we have constructed IF-VR, the first multimodal dataset of user implicit feedback in video recommendation scenario. This dataset covers two mainstream app-based recommendation modes: sequential video recommendation, where users can skip videos they dislike, and click simulation, which predicts click events on pages containing multiple videos and their titles. Specifically, IF-VR includes data from 15,000 users, covering their demographic profiles and interest tags, along with 933K clicking or watching histories. In addition, it contains 50K dislike feedback entries and 72K annotated types behind users' fast-swipes or low play rates in viewing histories. This dataset also includes 25K videos watched by users and their corresponding titles. Sourced from the Tencent Video Mobile app, IF-VR thus stands as a multimodal recommendation dataset that closely aligns with real-world scenarios. Experimental results demonstrate that our method outperforms other LLMs and MLLMs, excelling in predicting user fast-swipes and non-clicks, with relevant analyses provided. Our contributions are summarized below.

- We propose a novel **G-UBS** paradigm, which consists of two key agents named **UGM** and **UFM**, for integrating group profiles into RL-based user simulation fine-tuning. Our G-UBS eliminates the noise in the individual implicit feedback with group profile aiding, thereby enhancing the accuracy of user simulation.
- We introduce **IF-VR**, a large-scale multitask dataset tailored to analyze user implicit feedback in real-world multimodal recommendation scenarios, providing comprehensive evaluation.
- We have conducted extensive experiments on both real-world business scenarios and open-source datasets. The results demonstrate that our approach outperforms existing LLMs and MLLMs, with a 4.0% higher proportion of videos achieving a play rate $>30\%$ and 14.9% higher reasoning accuracy.

Related Work

Mining Implicit User Feedback. Currently, there are two main paradigms for mining users' implicit feedback: embedding-based and LLM-based methods. Embedding-based methods directly map implicit user feedback to embeddings (Xie et al. 2021; Chen et al. 2021; Paudel, Luck, and Bernstein 2018; Park and Lee 2022; Lai, Chen, and Zhang 2024) , which are incorporated into the recommendation pipeline. For instance, DFN(Xie et al. 2021) captures unbiased preferences using internal and external feedback, while CDR(Chen et al. 2021) uses explicit dislike signals to evaluate behavioral sequences. However, such embedding methods suffer from poor interpretability, offering limited insight into the underlying reasons for user dissatisfaction. LLM-based methods explore the use of LLMs to interpret implicit feedback via CoT reasoning (Yang et al. 2025b; Zhao, Xu, and Li 2025; Lai et al. 2025) or the RL training pipeline (Han et al. 2025; Yue et al. 2025; Lai et al. 2023). However, these single-modal methods do not consider the multimodal feedback noise in individual users.

User Simulator. Existing user simulation methods can be divided into two categories. One is system simulation in the context of recommendation systems (Zhang et al. 2025, 2024a; Zhao et al. 2023; Corecco et al. 2024; Zhang et al. 2024c; Wang et al. 2023; Chen et al. 2024; Wang et al. 2025b), which aims to mimic user interactions such as clicks, skips, or feedback. However, these single-modal frameworks cannot integrate multimodal signals prevalent in video recommendation scenarios. The other category is large-scale, LLM-driven multi-agent simulations (Yang et al. 2024b; Piao et al. 2025; Liu et al. 2025; Chen et al. 2025b,a,c; Lai et al. 2024), designed to reproduce emergent behaviors in open-ended social settings and offer insights into collective dynamics. However, this large-scale user simulation approach heavily relies on LLMs' capabilities. Despite simulating many users, LLMs lack accuracy in understanding users' implicit feedback without fine-tuning.

Method

This section details our Group-Aware User Behavior Simulation (G-UBS) paradigm, where two LLM-based agents (i.e., UGM and UFM) collaborate to integrate group profiles

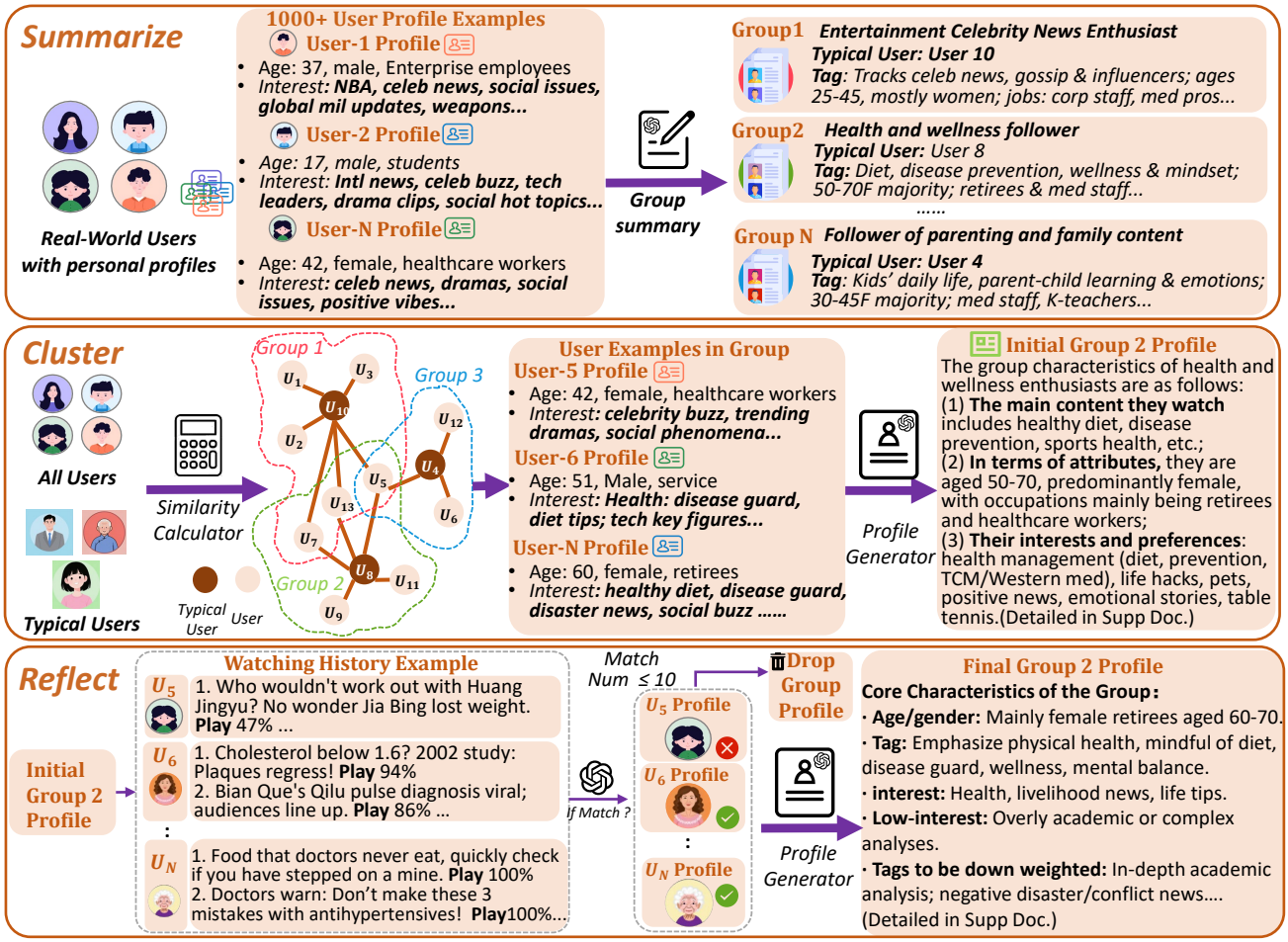

Figure 2: Overview of UGM group profile generation pipeline.

to ensure robust individual user simulations. Then, the construction methodology of the IF-VR dataset is presented.

Agent 1: User Group Manager (UGM)

To obtain robust and representative group profiles while achieving accurate user classification, we propose a summarize-cluster-reflect workflow. As illustrated in Fig. 2, UGM first performs initial classification on the user groups to be analyzed and summarizes preliminary group profiles. In the subsequent clustering phase, users are assigned to their respective groups. To mitigate deviations, during the reflection phase, UGM adjusts the prior clusters and outputs refined group profiles. Details are provided as follows. To be specific, the UGM's prompt is in Appendix A.1, detailed initial and final group profiles are in Appendix A.8.

Phase 1: Summarizing Initial Group Profiles. To establish an initial classification framework for large-scale user groups to be analyzed (supporting over 1,000 users), we first perform the "summarize" action. Given a user profile set $\mathcal{U}$ containing over 1000 users, each user is formatted as $u = [\text{ID}, \text{Occ}, \text{Age}, \text{Gender}, \mathbf{T}], u \in \mathcal{U}$. Here, Occ denotes the user's occupation and $\mathbf{T}$ represents tags for the user's video preferences as shown in Fig. 2. The set $\mathcal{U}$, the expected number of categories k, and the summary mode M are input into the group summary LLM $\mathcal{S}$ (DeepSeek-R1 (Guo et al. 2025)). The model outputs k user groups $\mathcal{G}$ along with their corresponding representative users U_g.The mode M specifies the grouping criteria, such as video preferences or demographic attributes (e.g., age, gender, occupation). This process is formulated as:

$$U_g, \mathcal{G} = \mathcal{S}(\mathcal{U}, k, M) \quad |\mathcal{G}| = k, |U_g| = k. \tag{1}$$

For each group $g \in \mathcal{G}$, $g = \{T_g, name\}$. T_g is the brief group description tag and $u_g \in U_g$ is the typical user of g.

Phase 2: Clustering users to Respective Groups. To accurately assign users to their respective groups, in this step, we cluster them by matching each user u to the typical users $u_g \in U_g$ across all categories based on similarity. The top 60 users most similar to each typical user u_g are selected to form the initial user group C_g, formulated as:

$$\mathcal{C}_g = \{u \in \mathcal{U}, u_g \in U_g \mid \text{Sim}(u, u_g) \geq \tau_g\}, g \in \mathcal{G} \tag{2}$$

The dynamic threshold (τ_j) is determined by ranking users in descending order of similarity and selecting the top 60. These users' profiles are then fed into the profile generator

Figure 3: Overview of UFM Reinforcement Learning Paradigm.

(GPT-4o) to create an initial group profile $\hat{P}_g$, as illustrated in the "cluster" section of Fig 2.

Phase 3: Refining Group Profiles via Reflection. Considering the context length constraints of LLMs, we did not incorporate user historical viewing records in the previous two working phases, which may lead to misclassifications. Therefore, in this step, we refine the initial clusters to rectify discrepancies between interest tags and actual viewing behavior. Taking group C_g as an example: for each user, their profiles $u \in C_g$ and the viewing history h (e.g., play rate, video title, duration, click) are input into the user-matching LLM (GPT-4o), which evaluates whether the user's interest and historical behavior align with $\hat{P}_g$. Users whose preferences match $\hat{P}_g$ are retained in the refined group C'_g.

$$C'_g = \left\{ u \in C_g \mid \text{Match}(u, \hat{P}_g, h) = \text{'Yes'} \right\} \quad (3)$$

For each category, if the number of matched users $|C'_g|$ is less than 10, no group profile is generated. Otherwise, all users' profiles and viewing histories within the group are input into the profile generator (GPT-4o) to summarize key information, including preference characteristics, comprehensive profile details, and recommendation directions, thereby forming the final group profile P_g as illustrated in Fig 2.

Agent 2: User Feedback Modeler (UFM)

We propose the UFM agent to robustly interpret users' implicit feedback under the guidance of relevant user group profiles, with Reinforcement Learning (RL) employed for model training. To enable the model to quickly grasp the core logic of the task and enhance the stability of RL, we first perform Supervised Fine-Tuning (SFT) following the approach of DeepSeek-R1 (Guo et al. 2025). Specifically, we leverage 50K collected explicit dislike feedback (e.g., dislike this content, dislike this author) and use GPT-4o (Hurst et al. 2024) to generate chain-of-thought annotations for attribution types. After warming up the model via SFT based on the above explicit dislike signals, we further train UFM following the RL paradigm presented in Fig 3, which consists of profile sampling and group-aware GRPO (GA-GRPO). Next, we will elaborate on these two components and the reward mechanism.

Profile Sampling. The UFM RL paradigm includes profile sampling and group-aware GRPO (GA-GRPO) as shown in Fig 3. GA-GRPO integrates group profiles and similar users into the policy model training, learning from both individual and group-level behaviors to enhance robustness and accuracy. To incorporate group knowledge into individual user simulators, profile sampling is executed at each tuning step, involving three types of profiles: (i) Training User Profile u_T: the primary user profile for training. (ii) Group profile P_g: a representative profile of the group to which u_T belongs. If u_T does not belong to any group, P_g is substituted with u_T. (iii) Similar User Profile u_S: a profile of another user from the same group(s) as u_T. As shown in Fig 3, if u_T belongs to both Group 1 and Group 2, u_s is sampled from other users in these two groups. If no similar user is available, u_s is replaced with u_T. We input the sampled profile to a User Feedback Modeler (UFM) that generates responses

Dataset	Age	Gender	Job	Interest Tag	Video Data	Explicit Feedback	Implicit Feedback	Negative Comments	Finish Rate	Click Rate	Play Rate
Amazon (Hou et al. 2024)	✗	✗	✗	✗	✗	✓	✗	✓	✗	✓	✗
Netflix (Bennett and Lanning 2007)	✗	✗	✗	✗	✗	✓	✓	✗	✓	✓	✗
Yelp (Asghar 2016)	✗	✗	✗	✗	✗	✓	✗	✓	✗	✓	✗
MIND (Wu et al. 2020)	✗	✗	✗	✓	✗	✗	✓	✗	✓	✓	✗
MovieLens (Harper and Konstan 2015)	✓	✓	✓	✓	✗	✓	✗	✗	✗	✓	✗
MircoLens (Ni et al. 2023)	✗	✗	✗	✗	✓	✗	✗	✗	✗	✓	✗
KuaiRand (Gao et al. 2022)	✗	✗	✗	✗	✗	✓	✓	✗	✓	✓	✗
IF-VR	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 1: Comparison with Previous Recommendation Dataset.

$O = \{o_S, o_T, o_G\}$. The UFM agent takes the sampled video V, its title, a sampled profile $u \in \{u_T, u_S\}$, and user viewing histories h_T and h_S as input, generating outputs o_T and o_S as $o = UFM(V, title, u, h)$, where $u \in \{u_T, u_S\}$ and $h \in \{h_T, h_S\}$. For the group profile P_g, only the sampled video frames V and title are fed into UFM, yielding $o_G = UFM(V, title, P_g)$.

Reward mechanism. To guide the model in understanding implicit user feedback, we categorize feedback into three types and train UFM using multiple-choice questions. To optimiz model performance, we design a reward mechanism that evaluates user behavior predictions based on specific criteria. For a given output $o \in O$, three types of rewards are considered: (i) Format reward: r_{format}: to ensure the model generates responses in the desired format (e.g., `<think>...</think>` for thoughts and `<answer>...</answer>` for answers), we introduce a format reward r_{format}. (ii) Skip reward: r_{skip}: if the model correctly predicts whether the user will fast-forward a video, r_{skip} is assigned. (iii) Choice reward: r_{choice}: if the model accurately chooses the reason options for fast-forwarding, r_{choice} is granted. The total reward is:

$$R(o) = r_{format} + r_{skip} + r_{choice}. \tag{4}$$

Group-Aware GRPO. To incorporate group-level and similar-user information, we weight the rewards derived from different profiles. Specifically, the rewards $\{R_T, R_S, R_G\}$ are calculated as $R_T = R(o_T) \times W_T$, $R_G = R(o_G) \times W_G$, and $R_S = R(o_S) \times W_S$, respectively. The quality of responses $o \in O$ is evaluated as:

$$A_R = \frac{R - \text{mean}(\{R_T, R_S, R_G\})}{\text{std}(\{R_T, R_S, R_G\})}, R \in \{R_T, R_S, R_G\}$$

A_R represents the relative quality of o. GA-GRPO optimizes for high-scoring responses while incorporating a KL divergence term to constrain the optimized policy π_θ from deviating excessively from the reference policy π_{ref}, where β serves as the regularization coefficient. This process is formulated as:

$$\max_{\pi_\theta} \mathbb{E}_{o \sim \pi_{\theta_{\text{old}}}(p)} \Big[\sum_{o \in O} \frac{\pi_\theta(o)}{\pi_{\theta_{\text{old}}}(o)} \cdot A_R - \beta \, \mathrm{D}_{\mathrm{KL}} \Big(\pi_\theta \,\|\, \pi_{\text{ref}} \Big) \Big]$$

The Prompt for UFM is in Appendix A.2. Further training details are provided in Appendix A.5.

Constructing the IF-VR Dataset

To address the scarcity of public multimodal datasets specifically designed for attributing users' implicit feedback, we constructed Video Recommendation Dataset with Implicit Feedback (IF-VR) to validate our G-UBS paradigm. IF-VR contains more user information and is closer to real-world recommendation scenarios, as shown in Tab 1.

Dataset Composition. As the first multimodal video dataset dedicated to implicit negative feedback verification, IF-VR includes 15K user profiles from the Tencent Video app, with detailed annotations such as gender, age, occupation shown in Tab 1. The viewing history spans 7 days from May 19 to May 25, 2025. This dataset covers two mainstream app-based recommendation modes: (i) Sequential video recommendation, allowing to skip disliked videos, with 8,000 users, 320K viewing histories, 50K explicit "dislike" feedback, and 72K implicit feedback annotations generated by GPT-4o and checked by humans. (ii) Click simulation, predicting user clicks on pages with multiple videos and titles, including 7,000 users, 613K exposure/click histories. In summary, IF-VR encompassed 15k users, 25K videos, and 993K interaction records. Appendix A.4 shows more details.

Labeling for IF-VR. To attribute these users' implicit feedback, we categorized the underlying causes into three types: **(i) Content-driven fast-skipping** (due to objective content flaws): e.g., vulgar content, clickbait titles, physiologically discomforting visuals (e.g., bloody scenes, unpleasant creatures like snakes or centipedes). **(ii) Algorithm-driven fast-skipping** (due to recommendation ineffectiveness): e.g., inaccurate user profiling, repetitive recommendations, insufficient diversity. **(iii) User-driven fast-skipping** (arising from individual user actions): such as operational errors and lack of viewing intent at the current time. To perform this categorization, we instructed GPT-4o to label the 72K histories with a viewing rate below 0.3 (from the 320K viewing entries) using the three types mentioned above, and double-checked by humans. More details are in Appendix A.3.

Experiments

Implementation Details and Metrics. In our experiments, we utilize Tencent Video APP's native recommendation system to generate a candidate video set, then apply our proposed G-UBS paradigm to filter out items mismatched with user preferences, yielding more precise out-

Model	Person Play Rate	Total Play Rate	Finish Rate	Play Rate >30%	Click Rate	Judge F1	Judge Acc	Reason F1	Reason Acc
Original Recommendation	46.5%	48.3%	17.1%	76.3%	21.4%	-	-	-	-
SASRec (Kang and McAuley 2018)	45.8%	48.0%	16.9%	75.8%	20.9%	-	-	-	-
Llama3.3-70b (Grattafiori et al. 2024)	46.5%	48.4%	17.3%	77.0%	21.6%	18.1%	71.9%	10.3%	8.6%
Qwen3-32b (Yang et al. 2025a)	47.8%	49.6%	17.6%	80.0%	21.8%	36.0%	64.1%	36.5%	35.1%
Qwen3-235b (Yang et al. 2025a)	48.3%	51.6%	18.3%	83.8%	21.9%	44.3%	65.9%	38.6%	42.3%
Deepseek-r1-0528 (Guo et al. 2025)	49.6%	53.0%	20.9%	83.8%	22.7%	43.0%	61.0%	41.3%	48.0%
Qwen-2.5VL-7B (Yang et al. 2024a)	47.2%	49.0%	17.1%	79.3%	22.0%	35.8%	55.6%	30.6%	36.4%
Video-R1 (Feng et al. 2025)	48.3%	49.1%	17.6%	80.4%	22.2%	36.7%	57.9%	31.7%	36.9%
Videochat-R1 (Li et al. 2025)	48.4%	52.5%	19.7%	84.5%	22.1%	42.4%	42.9%	42.0%	41.6%
Doubao-1.5-pro (Shen et al. 2025)	48.8%	50.0%	18.4%	81.0%	22.8%	38.6%	70.5%	16.5%	22.5%
GPT-4o (Hurst et al. 2024)	51.3%	52.8%	19.9%	84.7%	23.0%	45.4%	65.1%	37.4%	40.5%
G-UBS	**52.3%**	**55.3%**	**22.1%**	**88.7%**	**25.7%**	**54.9%**	**72.9%**	**55.6%**	**62.9%**

Table 2: SOTA Comparison on Different LLM and MLLM in Understanding the Implicit Feedback of Users

	MovieLens			Amazon Books		
Method	**Acc**	**Recall**	**F1**	**Acc**	**Recall**	**F1**
RecAgent	58.1%	60.4%	62.1%	62.7%	64.9%	65.0%
Agent4Rec	69.1%	69.1%	69.8%	68.9%	70.3%	67.9%
GPT-4o	72.2%	71.8%	73.6%	73.4%	72.8%	73.6%
SimUser	79.1%	75.8%	77.7%	79.1%	78.5%	79.4%
G-UBS	**79.9%**	**76.2%**	**78.2%**	**80.1%**	**78.9%**	**80.2%**

Table 3: User Simulation Experiment on Public Datasets.

Group Num	Person Play Rate	Play Rate >30%	Click Rate	Judge F1	Reason F1
10	52.1%	88.5%	25.5%	54.5%	55.1%
20	**52.3%**	**88.7%**	**25.7%**	**54.9%**	**55.6%**
30	52.3%	88.6%	25.4%	54.6%	55.3%
40	52.3%	88.7%	25.2%	54.3%	55.4%

Table 4: Different Grouping Numbers on UGM

comes. The LLM employed throughout the G-UBS workflow is Qwen2.5-VL-7B (Bai et al. 2025). We perform full-parameter fine-tuning on the UFM agent using 4 A100 80G GPUs. For both SFT and RFT, the learning rate is set to 1e-5. We conducted SFT for 1 epoch and RFT for 200 steps to achieve optimal results on IF-VR. Additional training details and experimental settings for IF-VR, MovieLens, and Amazon Books are provided in Appendix A.6 and A.7. The evaluation metrics in Tab 2 are defined as follows: Play Rate is the ratio of watch time to video duration, with repeated views counted as 100%. Person Play Rate and Total Play Rate refer to the average Play Rate per user and across all videos, respectively. Finish Rate is the proportion of videos with Play Rate>90%, and Play Rate>30% is the percentage of videos with a Play Rate>30%. Click Rate is the ratio of clicks to recommended videos. Judge F1/Acc evaluate the model's ability to predict skips, while Reason F1/Acc assess its accuracy in identifying why users skip a video. Following SimUser (Xiang et al. 2024), the metrics (ACC, Recall, F1) in Tab 3 measure the model's ability to predict whether users will like a video.

Interest	Demographics	Person Play Rate	Play Rate >30%	Click Rate	Judge F1	Reason F1
✓	✗	**52.3%**	**88.7%**	**25.7%**	**54.9%**	**55.6%**
✗	✓	52.0%	88.3%	25.6%	54.6%	55.1%
✓	✓	52.2%	88.7%	25.4%	54.8%	55.3%

Table 5: Different Grouping Methods on UGM

Model	Person Play Rate	Play Rate >30%	Click Rate	Judge F1	Reason F1
BERT-Sim	51.8%	88.0%	24.9%	54.3%	54.7%
K-Means	51.6%	87.9%	25.0%	54.6%	55.0%
TFIDF	**52.3%**	**88.7%**	**25.7%**	**54.9%**	**55.6%**

Table 6: Different User Clustering Methods on UGM

SFT	RL	Group Profile	Person Play Rate	Play Rate >30%	Click Rate	Judge F1	Reason F1
✗	✗	✗	47.2%	79.3%	22.8%	35.8%	30.6%
✓	✗	✗	48.6%	80.8%	23.6%	38.0%	36.4%
✗	✓	✗	49.6%	84.7%	23.2%	45.3%	46.0%
✓	✓	✗	51.2%	87.4%	23.8%	51.4%	46.5%
✗	✓	✓	50.8%	87.9%	24.9%	52.6%	50.0%
✓	✓	✓	**52.3%**	**88.7%**	**25.7%**	**54.9%**	**55.6%**

Table 7: Training Processes on UFM

Comparison with SOTAs. As shown in Tab 2, we compare our method with Tencent Video APP's original recommendation strategy, methods based on LLMs (Grattafiori et al. 2024; Yang et al. 2025a; Guo et al. 2025) and MLLMs (Yang et al. 2024a; Feng et al. 2025; Shen et al. 2025; Hurst et al. 2024; Li et al. 2025), and the traditional item-based SASRec (Kang and McAuley 2018). Experimental results show that our post-processed recommendation strategy achieves the best overall performance. Traditional methods, such as SASRec, rely solely on item embeddings and fail to grasp concrete item semantics. Meanwhile, LLMs and MLLMs poorly align with user preferences due to inherent noise in user feedback. Specifically, our method

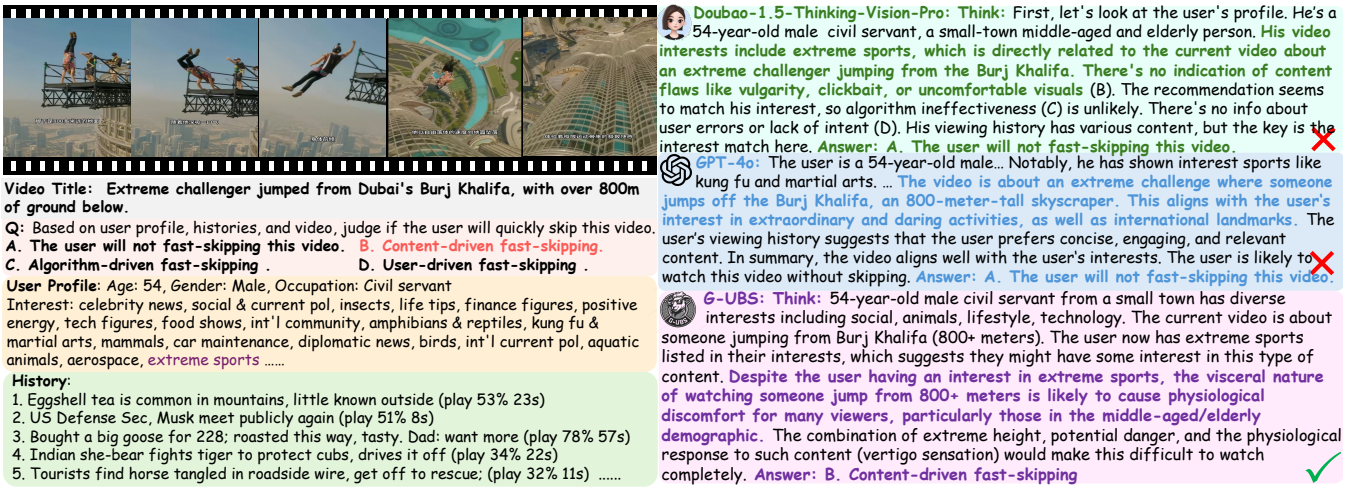

Figure 4: Case Study of G-UBS.

significantly boosts user engagement: Person Play Rate increases from 46.5% to 52.3%, and Total Play Rate rises from 48.3% to 55.3%, outperforming top LLM (Deepseek-R1) and MLLM (GPT-4o) baselines, attributing to learning from different users within groups. Moreover, our model achieves the highest Reasoning Accuracy and F1 scores, indicating a strengthened capability to interpret users' implicit feedback. These results confirm the G-UBS paradigm's effectiveness.

User Simulation Experiments on Public Datasets. To verify G-UBS's applicability, we compared it with RecAgent (Wang et al. 2025a), Agent4Rec (Zhang et al. 2024b), GPT-4o (Hurst et al. 2024), and SimUser (Xiang et al. 2024) using MovieLens and Amazon Books dataset, following SimUser (Xiang et al. 2024)'s approach. Tab 3 reports the binary classification accuracy of this experiment, which measures the model's ability to simulate user click responses. Our G-UBS achieves the highest simulation accuracy.

Ablation Study

Different Grouping Methods on UGM. We ablate the optimal grouping strategies for the 'summarize' process as shown in Tab 5. Interest-based grouping outperforms both demographic-only and hybrid (interest+demographic) approaches. This is because demographic grouping (by age, gender, or occupation) suffers from large intra-group interest divergence. Users sharing the same attributes often have different video preferences and opinions, reducing the consistency of group profiles.

Different User Clustering Methods on UGM. We conduct various clustering methods to determine the optimal choice in the 'cluster' process of UGM. In Tab 6, TF-IDF outperforms BERT and traditional ML methods. Since user profiles are structured as concatenated word sequences rather than natural language texts, TF-IDF outperforms BERT (which excels at sentence understanding) and K-Means (which lacks semantic understanding) in word matching.

Training Processes on UFM. We conduct ablation studies on the UFM training process(with Qwen2.5-VL as the baseline) to analyze the impacts of SFT, RL, and group profile. As shown in Tab 7, SFT and RL boost performance incrementally. But adding group profiles, especially SFT+RL+Group, drives the best results. This result validates the efficacy of our UFM agent tuning pipeline.

Hyperparameter Analysis

Different Grouping Numbers on UGM. We vary the number of groups in the 'summarize' process of UGM to determine the optimal granularity. The results from Table 4 shows that 20 groups yield the best performance (e.g., 52.3% Person Play Rate). Fewer groups lead to significant intra-group interest divergence, which undermines the consistency of group profiles. Conversely, an excessive number of groups reduces the number of users per group, resulting in group profiles that only serve a small minority and lack representativeness, thereby leading to suboptimal results.

Visualization of G-UBS. Regarding the video of jumping off the Burj Khalifa in Fig.4, both Doubao and GPT-4o concluded that the user would not fast-skip it based on the user's profile. However, such high-altitude extreme sports videos can cause dizziness and physical discomfort, especially for middle-aged and elderly people. Our G-UBS can accurately determine why the user would fast-skip it based on the group characteristics of middle-aged and elderly people.

Conclusion

In this paper, we propose a novel Group-aware User Behavior Simulation (G-UBS) paradigm that captures and leverages group profiles to achieve a more robust and profound understanding of users' implicit feedback. We have also constructed a large-scale IF-VR dataset to support experimental evaluations. Compared with mainstream LLMs and MLLMs, G-UBS achieves a 4.0% higher proportion of videos with a play rate exceeding 30% and a 14.9% improvement in reasoning accuracy on the IF-VR dataset. Extensive experiments validate the efficacy of our proposed method.

Acknowledgements

This work was supported by the National Key R&D Program of China (NO.2022ZD0160505).

References

Asghar, N. 2016. Yelp dataset challenge: Review rating prediction. *arXiv preprint arXiv:1605.05362*.

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; et al. 2025. Qwen2. 5-vl technical report. *arXiv preprint arXiv:2502.13923*.

Bennett, J.; and Lanning, S. 2007. The netflix prize.

Chen, B.; Chen, S.; Li, K.; Xu, Q.; Qiao, Y.; and Wang, Y. 2024. Percept, chat, and then adapt: Multimodal knowledge transfer of foundation models for open-world video recognition. *arXiv preprint arXiv:2402.18951*.

Chen, B.; Chen, S.; Li, K.; Xu, Q.; Qiao, Y.; and Wang, Y. 2025a. Super Encoding Network: Recursive Association of Multi-Modal Encoders for Video Understanding. *arXiv preprint arXiv:2506.07576*.

Chen, B.; Yue, Z.; Chen, S.; Wang, Z.; Liu, Y.; Li, P.; and Wang, Y. 2025b. Lvagent: Long video understanding by multi-round dynamical collaboration of mllm agents. *arXiv preprint arXiv:2503.10200*.

Chen, H.; Chen, Y.; Wang, X.; Xie, R.; Wang, R.; Xia, F.; and Zhu, W. 2021. Curriculum disentangled recommendation with noisy multi-feedback. *Advances in Neural Information Processing Systems*, 34: 26924–26936.

Chen, S.; Chen, B.; Yu, C.; Luo, Y.; Yi, O.; Cheng, L.; Zhuo, C.; Li, Z.; and Wang, Y. 2025c. VRAgent-R1: Boosting Video Recommendation with MLLM-based Agents via Reinforcement Learning. *arXiv preprint arXiv:2507.02626*.

Corecco, N.; Piatti, G.; Lanzendörfer, L. A.; Fan, F. X.; and Wattenhofer, R. 2024. SUBER: An RL Environment with Simulated Human Behavior for Recommender Systems. *arXiv preprint arXiv:2406.01631*.

Feng, K.; Gong, K.; Li, B.; Guo, Z.; Wang, Y.; Peng, T.; Wu, J.; Zhang, X.; Wang, B.; and Yue, X. 2025. Video-r1: Reinforcing video reasoning in mllms. *arXiv preprint arXiv:2503.21776*.

Gao, C.; Li, S.; Zhang, Y.; Chen, J.; Li, B.; Lei, W.; Jiang, P.; and He, X. 2022. Kuairand: An unbiased sequential recommendation dataset with randomly exposed videos. In *Proceedings of the 31st ACM international conference on information & knowledge management*, 3953–3957.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Guo, D.; Yang, D.; Zhang, H.; Song, J.; Zhang, R.; Xu, R.; Zhu, Q.; Ma, S.; Wang, P.; Bi, X.; et al. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Guo, H.; Tang, R.; Ye, Y.; Li, Z.; and He, X. 2017. DeepFM: a factorization-machine based neural network for CTR prediction. *arXiv preprint arXiv:1703.04247*.

Han, E.; Chen, J.; Sankararaman, K. A.; Peng, X.; Xu, T.; Helenowski, E.; Peng, K.; Kumar, M.; Wang, S.; Fang, H.; et al. 2025. Reinforcement Learning from User Feedback. *arXiv preprint arXiv:2505.14946*.

Harper, F. M.; and Konstan, J. A. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4): 1–19.

He, R.; Fang, C.; Wang, Z.; and McAuley, J. 2016. Vista: A visually, socially, and temporally-aware model for artistic recommendation. In *Proceedings of the 10th ACM conference on recommender systems*, 309–316.

Hou, Y.; Li, J.; He, Z.; Yan, A.; Chen, X.; and McAuley, J. 2024. Bridging language and items for retrieval and recommendation. *arXiv preprint arXiv:2403.03952*.

Hurst, A.; Lerer, A.; Goucher, A. P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; Hayes, A.; Radford, A.; et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Kang, W.-C.; and McAuley, J. 2018. Self-attentive sequential recommendation. In *2018 IEEE international conference on data mining (ICDM)*, 197–206. IEEE.

Lai, R.; Chen, L.; Chen, R.; and Zhang, C. 2025. DAR: Dimension-Adaptive Recommendation with Multi-Granular Noise Control. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2203–2212.

Lai, R.; Chen, L.; Zhao, Y.; Chen, R.; and Han, Q. 2023. Disentangled negative sampling for collaborative filtering. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 96–104.

Lai, R.; Chen, R.; Han, Q.; Zhang, C.; and Chen, L. 2024. Adaptive hardness negative sampling for collaborative filtering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 8645–8652.

Lai, R.; Chen, R.; and Zhang, C. 2024. A survey on data-centric recommender systems. *arXiv preprint arXiv:2401.17878*.

Li, X.; Yan, Z.; Meng, D.; Dong, L.; Zeng, X.; He, Y.; Wang, Y.; Qiao, Y.; Wang, Y.; and Wang, L. 2025. Videochat-r1: Enhancing spatio-temporal perception via reinforcement fine-tuning. *arXiv preprint arXiv:2504.06958*.

Liu, G.; Le, V.; Rahman, S.; Kreiss, E.; Ghassemi, M.; and Gabriel, S. 2025. Mosaic: Modeling social ai for content dissemination and regulation in multi-agent simulations. *arXiv preprint arXiv:2504.07830*.

Ni, Y.; Cheng, Y.; Liu, X.; Fu, J.; Li, Y.; He, X.; Zhang, Y.; and Yuan, F. 2023. A content-driven micro-video recommendation dataset at scale. *arXiv preprint arXiv:2309.15379*.

Park, M.; and Lee, K. 2022. Exploiting negative preference in content-based music recommendation with contrastive learning. In *Proceedings of the 16th ACM Conference on Recommender Systems*, 229–236.

Paudel, B.; Luck, S.; and Bernstein, A. 2018. Loss aversion in recommender systems: Utilizing negative user preference to improve recommendation quality. *arXiv preprint arXiv:1812.11422*.

Piao, J.; Yan, Y.; Zhang, J.; Li, N.; Yan, J.; Lan, X.; Lu, Z.; Zheng, Z.; Wang, J. Y.; Zhou, D.; et al. 2025. Agentsociety: Large-scale simulation of llm-driven generative agents advances understanding of human behaviors and society. *arXiv preprint arXiv:2502.08691*.

Shen, W.; Liu, G.; Wu, Z.; Zhu, R.; Yang, Q.; Xin, C.; Yue, Y.; and Yan, L. 2025. Exploring data scaling trends and effects in reinforcement learning from human feedback. *arXiv preprint arXiv:2503.22230*.

Wang, L.; Zhang, J.; Yang, H.; Chen, Z.-Y.; Tang, J.; Zhang, Z.; Chen, X.; Lin, Y.; Sun, H.; Song, R.; et al. 2025a. User behavior simulation with large language model-based agents. *ACM Transactions on Information Systems*, 43(2): 1–37.

Wang, X.; Tang, X.; Zhao, W. X.; Wang, J.; and Wen, J.-R. 2023. Rethinking the evaluation for conversational recommendation in the era of large language models. *arXiv preprint arXiv:2305.13112*.

Wang, Z.; Chen, B.; Yue, Z.; Wang, Y.; Qiao, Y.; Wang, L.; and Wang, Y. 2025b. VideoChat-A1: Thinking with Long Videos by Chain-of-Shot Reasoning. *arXiv preprint arXiv:2506.06097*.

Wu, F.; Qiao, Y.; Chen, J.-H.; Wu, C.; Qi, T.; Lian, J.; Liu, D.; Xie, X.; Gao, J.; Wu, W.; et al. 2020. Mind: A large-scale dataset for news recommendation. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, 3597–3606.

Xiang, W.; Zhu, H.; Lou, S.; Chen, X.; Pan, Z.; Jin, Y.; Chen, S.; and Sun, L. 2024. SimUser: Generating Usability Feedback by Simulating Various Users Interacting with Mobile Applications. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 1–17.

Xie, R.; Ling, C.; Wang, Y.; Wang, R.; Xia, F.; and Lin, L. 2021. Deep feedback network for recommendation. In *Proceedings of the twenty-ninth international conference on international joint conferences on artificial intelligence*, 2519–2525.

Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025a. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Yang, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Li, C.; Liu, D.; Huang, F.; Wei, H.; et al. 2024a. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Yang, S.; Cao, J.; Li, H.; Mao, Y.; and Pang, S. 2025b. RecCoT: Enhancing Recommendation via Chain-of-Thought. *arXiv preprint arXiv:2506.21032*.

Yang, Z.; Zhang, Z.; Zheng, Z.; Jiang, Y.; Gan, Z.; Wang, Z.; Ling, Z.; Chen, J.; Ma, M.; Dong, B.; et al. 2024b. Oasis: Open agent social interaction simulations with one million agents. *arXiv preprint arXiv:2411.11581*.

Yue, Z.; Zhang, H.; Zeng, X.; Chen, B.; Wang, C.; Zhuang, S.; Dong, L.; Du, K.; Wang, Y.; Wang, L.; et al. 2025. UniFlow: A Unified Pixel Flow Tokenizer for Visual Understanding and Generation. *arXiv preprint arXiv:2510.10575*.

Zhang, A.; Chen, Y.; Sheng, L.; Wang, X.; and Chua, T.-S. 2024a. On generative agents in recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in Information Retrieval*, 1807–1817.

Zhang, A.; Chen, Y.; Sheng, L.; Wang, X.; and Chua, T.-S. 2024b. On generative agents in recommendation. In *Proceedings of the 47th international ACM SIGIR conference on research and development in Information Retrieval*, 1807–1817.

Zhang, E.; Wang, X.; Gong, P.; Lin, Y.; and Mao, J. 2024c. Usimagent: Large language models for simulating search users. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2687–2692.

Zhang, Z.; Liu, S.; Liu, Z.; Zhong, R.; Cai, Q.; Zhao, X.; Zhang, C.; Liu, Q.; and Jiang, P. 2025. Llm-powered user simulator for recommender system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 13339–13347.

Zhao, K.; Liu, S.; Cai, Q.; Zhao, X.; Liu, Z.; Zheng, D.; Jiang, P.; and Gai, K. 2023. KuaiSim: A comprehensive simulator for recommender systems. *Advances in Neural Information Processing Systems*, 36: 44880–44897.

Zhao, K.; Xu, F.; and Li, Y. 2025. Reason-to-Recommend: Using Interaction-of-Thought Reasoning to Enhance LLM Recommendation. *arXiv preprint arXiv:2506.05069*.

Zhao, X.; Zhang, L.; Ding, Z.; Xia, L.; Tang, J.; and Yin, D. 2018. Recommendations with negative feedback via pairwise deep reinforcement learning. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 1040–1048.