LLM-Empowered Creator Simulation for Long-Term Evaluation of Recommender Systems Under Information Asymmetry

Xiaopeng Ye
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
xpye@ruc.edu.cn

Chen Xu
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
xc_chen@ruc.edu.cn

Zhongxiang Sun
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
sunzhongxiang@ruc.edu.cn

Jun Xu*
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
junxu@ruc.edu.cn

Gang Wang
Zhenhua Dong
Huawei Noah's Ark Lab
Shenzhen, China
{wanggang110,dongzhenhua}@huawei.com

Ji-Rong Wen
Gaoling School of Artificial Intelligence
Renmin University of China
Beijing, China
jrwen@ruc.edu.cn

Abstract

Maintaining the long-term sustainability of recommender systems (RS) is crucial. Traditional RS evaluation methods primarily focus on the user's immediate feedback (e.g., click), however, they often overlook the long-term effect involved by the content creators. In the real world, content creators can strategically create and upload new items to the platform by analyzing users' feedback and preference trends. Although previous studies have attempted to model creator behaviors, they often overlook that such behaviors are under conditions of *information asymmetry*. This asymmetry arises because creators mainly access the user feedback on the items they produce, while the platform has access to the full spectrum of feedback data. However, existing RS simulators often fail to consider such a condition, making the long-term RS evaluation inaccurate.

To bridge this gap, we propose a Large Language Model (LLM)-empowered creator simulation agent named **CreAgent**. By utilizing the belief mechanism from game theory and the fast-and-slow thinking framework, we can simulate the creator's behaviors well under information asymmetry. Furthermore, to enhance CreAgent's simulation ability, we utilize Proximal Policy Optimization to fine-tune CreAgent. Our credibility validation experiments demonstrate that our simulation environment effectively aligns with the behaviors of real-world platforms and creators, thereby enhancing the reliability of long-term evaluations in RS. Furthermore, leveraging this simulator, we can examine whether RS algorithms, such as fairness- and diversity-aware methods, contribute to improving long-term performance for different stakeholders.

*Corresponding author. Work partially done at Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education.

SIGIR '25, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730026

CCS Concepts

• **Information systems → Recommender systems**.

Keywords

Long-term Evaluation, Content Creator, Information Asymmetry, LLM-empowered Agent

ACM Reference Format:
Xiaopeng Ye, Chen Xu, Zhongxiang Sun, Jun Xu, Gang Wang, Zhenhua Dong, and Ji-Rong Wen. 2025. LLM-Empowered Creator Simulation for Long-Term Evaluation of Recommender Systems Under Information Asymmetry. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy*. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3726302.3730026

1 Introduction

Recently, improving the long-term sustainability of multi-stakeholder recommendation platforms [1, 6], such as YouTube and TikTok, has emerged as a critical concern [14, 42, 49]. Traditional evaluation methods for recommender systems (RS) typically focus on metrics that measure the prediction accuracy of immediate user responses (e.g., clicks, purchases) [19, 65]. However, these short-term evaluations often fail to account for long-term effects (e.g., creator retention rate [12], long-term user engagement [72]), leading to potential risks for platform developments, such as the emergence of filter bubbles [37].

Creator behaviors under information asymmetry. When evaluating the long-term impact of RS, it is crucial to consider the content creators [4, 6]. This is because (1) creators continuously reshape the platform's content ecosystem by uploading items over time; and (2) RS can significantly influence creators' behavior, which in turn impacts the platform's long-term development [2]. While previous studies have attempted to model the interaction behavior between creators and RS [10, 59, 70], they often overlook the fact that such interaction behaviors are acutally under the condition of **information asymmetry** [3, 40, 60]. This asymmetry arises because creators typically have access only to feedback on the items they produced, as restricted by platform policies [13], while the

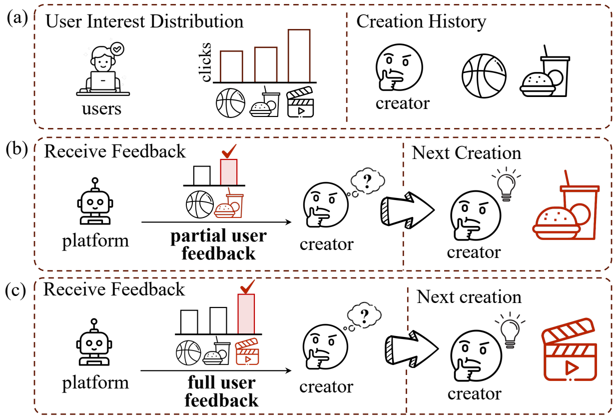


Figure 1: (a) A platform where users favor movies, food, and sports in decreasing order, with a creator who has created sports and food items. Comparison of creator behavior under two conditions: (b) limited information and (c) a hypothetical scenario with full information.

platform has access to the full spectrum of feedback data. To better illustrate this, we will provide an example as follows.

As shown in Figure 1 (a), we consider a platform with creators, users, and three genres of items. On such a platform, we assume that the majority of users prefer the genre of movies, followed by a preference for food-related items and then sports-related items. In a real-world scenario, the platform will only share user feedback on items created by a specific creator directly with that creator. Under such partial feedback, the creator that produced sports and food item categories will be more likely to create the new food rather than sports items. This is because the food receives more positive feedback from users (as illustrated in Figure 1 (b)). However, if the platform provided him with full user feedback information hypothetically (as illustrated in Figure 1 (c)), the creator would likely opt to create a movie item instead. Since the movie is welcomed by most users in the platforms. Therefore, modeling the information asymmetry is important for simulating the creator's strategic behaviors. It is also highlighted by existing mechanism design studies [3, 40].

Simulator for RS long-term evaluation. Although online test [23] can effectively assess the performance of RS in environments with creators, its high costs hinder the evaluation efficiency [65]. Recognizing the need for cost-efficient solutions, recent studies [16, 53, 67] have identified recommendation simulators as an ideal approach for affordable long-term evaluation. However, existing simulators [16, 46, 67] often focus exclusively on user behavior simulation, neglecting the impact of creators.

To achieve realistic creator behavior simulations under platform-creator information asymmetry scenarios, we propose a Large Language Model (LLM)-empowered creator agent, called **CreAgent**. Inspired by the game theory [40], we employ a belief module to reflect the limited information status of creators and utilize the fast-and-slow thinking mechanism [20] to capture the thought process under such information asymmetry. Moreover, to better help CreAgent understand the limited user feedback information, we utilize Proximal Policy Optimization (PPO) [43] to fine-tune the CreAgent. Credibility validation experiments have confirmed CreAgent's ability to exhibit human-like strategic creator behaviors, supported by insights from real-world data patterns and behavior economics principles [9, 21].

Equipped with the CreAgent, we design a simulated platform environment, which contains highly extensible recommendation models and modified widely-used user agents RecAgent [53]. The simulated platform environment is built using a large-scale dataset that we collected through web crawling from the YouTube website. Based on such a dataset, our experiments confirm that our proposed CreAgent can well align with the behaviors of real content creators under information asymmetry interaction patterns. Furthermore, utilizing this simulation environment, we reassess various fairness- and diversity-aware RS algorithms to analyze their long-term performance for users, creators, and the platform separately.

Contributions. Our contribution can be summarized as follows:

(1) We emphasize the significance of considering creator strategic behavior under platform-creator information asymmetry scenario during the long-term evaluation of RS.

(2) We introduce CreAgent, an LLM-empowered agent that simulates real-world creator interactions, and an extensible simulation platform for evaluating long-term RS performance with diverse models and user agents.

(3) Extensive experiments demonstrated that our simulation platform effectively replicates real-world RS, enabling a thorough reassessment of the long-term performance of various RS models.

2 Related Work

Evaluation of RS. How to evaluate RS is a complex and essential task, which can be divided into short-term and long-term evaluation based on their objectives [65]. Most current studies focus on the short-term objective using offline metrics [17, 18], relying on pre-collected log data containing users' explicit (e.g., ratings) or implicit (e.g., click) feedback to compute metrics like prediction accuracy [57] and ranking metrics [19]. Due to the inefficiency and high cost of long-term online A/B test [23, 42], offline long-term evaluation has gained significant attention in recent years, which can be divided into two main categories: (1) use short-term metric [14] or data [42] to predict long-term performance, (2) create an RS simulator to replicate the real-world environment [16, 53, 67] for evaluation. In this paper, we focus on the second type.

RS simulator for long-term evaluation. Most existing recommendation simulators (e.g., LLM-based simulator [53, 67], reinforcement learning (RL)-based simulator [16, 46]) focus on user simulation while overlooking creators, making it difficult to capture the long-term dynamic of content platforms. Some data-driven methods are proposed to conduct creator simulation. SimuLine [69] applied heuristic methods to determine creators' next creation in news recommendation. Some works [32, 33] assumed that creators will leave the platform if their user engagement falls below a certain threshold. Other modeling methods used user positive feedback (e.g., click) as the gradient to update the creation state [29, 63, 66]. However, these approaches failed to align with real creation behavior because: (1) they are unable to produce authentic content (e.g., text), instead relying on embeddings to represent the content they create [69]; (2) they cannot capture the personalization of real-world creators; (3) they ignored human behavior patterns under information asymmetry, such as risk aversion in prospect theory [21] and bounded rationality [44].

Behaviors under information asymmetry. Creator behaviors in information asymmetry conditions have been actively studied and emphasized in the game theory literature [29, 60, 63]. They typically assume that creators are totally rational, i.e., aiming to maximize their utility, which often lacks personalization and differs from real-world human behavior under risk (i.e., bounded rational [44]). Rule-based [60] and heuristic method [29, 63] are applied to model the strategic behavior. These studies mainly focus on the competition among creators [29, 63] and the design of better platform mechanisms[32, 40] to maximize user welfare.

3 Preliminaries

Simulation platform. In RS, we use $\mathcal{P} = \{\mathcal{U}, \mathcal{I}, \mathcal{C}, \mathcal{G}, \mathcal{D}, \mathbf{E}, \mathbf{Y}\}$ to denote the information known to the platform. $\mathcal{U} = \{u\}$, $\mathcal{I} = \{i\}$, $\mathcal{C} = \{c\}$, $\mathcal{G} = \{g\}$ is the set of users, items, creators, and item genres. Each item $i \in \mathcal{I}_g$ in RS belongs to a genre $g \in \mathcal{G}$. An item i created by creator c denotes as $i \in \mathcal{I}_c$. $\mathbf{E}$, $\mathbf{Y}$ is the user-item exposure matrix and interaction matrix.

At each time period $t \in [1, 2, \cdots, T]$, some users $\mathcal{U}^n$ will visit the RS, and some active creators $\mathcal{C}^t$ will create items. Each created item i will be added to the item pool $\mathcal{I}$ of RS. The items created in time period t are denoted as $i \in \mathcal{I}^t$. For each user u, the RS generates a list of K recommended items chosen from $\mathcal{I}$. Each user u can choose to click or skip these items. Then, the RS records the click and exposure interaction in $\mathbf{Y}(t)$ and $\mathbf{E}(t)$, which denotes the exposure matrix and interaction matrix of time period t. Specifically, $\mathbf{Y}_{u,i}(t) = 1$ if user u clicks item i at time period t and 0 otherwise, $\mathbf{E}_{u,i}(t) = 1$ if item i is exposed to user u at time period t and 0 otherwise. Also, RS will save the interaction record $(u, i, \mathbf{Y}_{u,i}(t))$ into the interaction record $\mathcal{D}$. For every T_{train} period, the RS will be re-trained using interactions in $\mathcal{D}$.

At the beginning of our simulation, we utilize the real-world RS dataset $\mathcal{P}^0 = \{\mathcal{U}^0, \mathcal{I}^0, \mathcal{C}^0, \mathcal{G}^0, \mathcal{D}^0\}$ to initialize the simulation environment, including the profile and initial memory/belief of users and creator agents. The initial RS is trained using the real-world interactions $\mathcal{D}^0$. During simulation, we use the first $t \in [1, 2, \cdots, T_0 - 1]$ time periods to initialize a stable RS simulation environment that reflects the real-world platform. Then, the subsequent $t \in [T_0, \cdots, T]$ periods are used to evaluate the long-term impact of different RS models.

Information asymmetry. In real-world platforms, such as YouTube, the platform can access comprehensive user behavior data (e.g., watch time, click/like history), policies [48] and incentives [40], which is not disclosed to creators. Meanwhile, creators can only infer whether their content aligns with user preferences by analyzing feedback (e.g., comments, likes, and other interactions) on their created items and then adjust their creation strategies accordingly. Thus, significant information asymmetry exists in the platforms, as highlighted by previous studies [29, 40, 63].

In this paper, we denote the platform-possessed information as $\mathcal{P}$, whereas each creator c has access to a subset of this information, denoted as $\mathcal{P}_c = \{\mathcal{I}_c, \mathbf{E}_i^\top, \mathbf{Y}_i^\top \mid i \in \mathcal{I}_c\}, \forall c \in \mathcal{C}$. Since $\mathcal{P}_c \subset \mathcal{P}$, this disparity indicates an information asymmetry between the platform and the creators.

Organization of the paper. To evaluate the long-term impact of RS under the platform-creator information asymmetry environment, we propose a simulation platform with creator behavior modeling. The overview of the simulation platform is illustrated in Figure 2. The simulation platform can be decoupled into the LLM-empowered creator agent **CreAgent** (see Section 4 and Figure 2 (a)) and the **platform environment** (see Section 5 and Figure 2 (b)). Afterwards, we conduct experiments to demonstrate the credibility of our proposed CreAgent (see Section 6). Finally, we use CreAgent and the simulation platform to assess the long-term impact of RS models on diverse stakeholders (see Section 7).

4 CreAgent

In this section, we leverage the human-like analyzing, generation capabilities, and world knowledge [50] of LLMs to simulate real-world strategic behavior of creators under information asymmetry scenarios. As demonstrated in Figure 2 (a), CreAgent consists of four modules: the profile and memory module, the game-theory-inspired belief module, and the creation module integrating fast-and-slow thinking [20]. Finally, we utilize PPO-based fine-tuning to enhance CreAgent's analytical and creative capabilities under limited information.

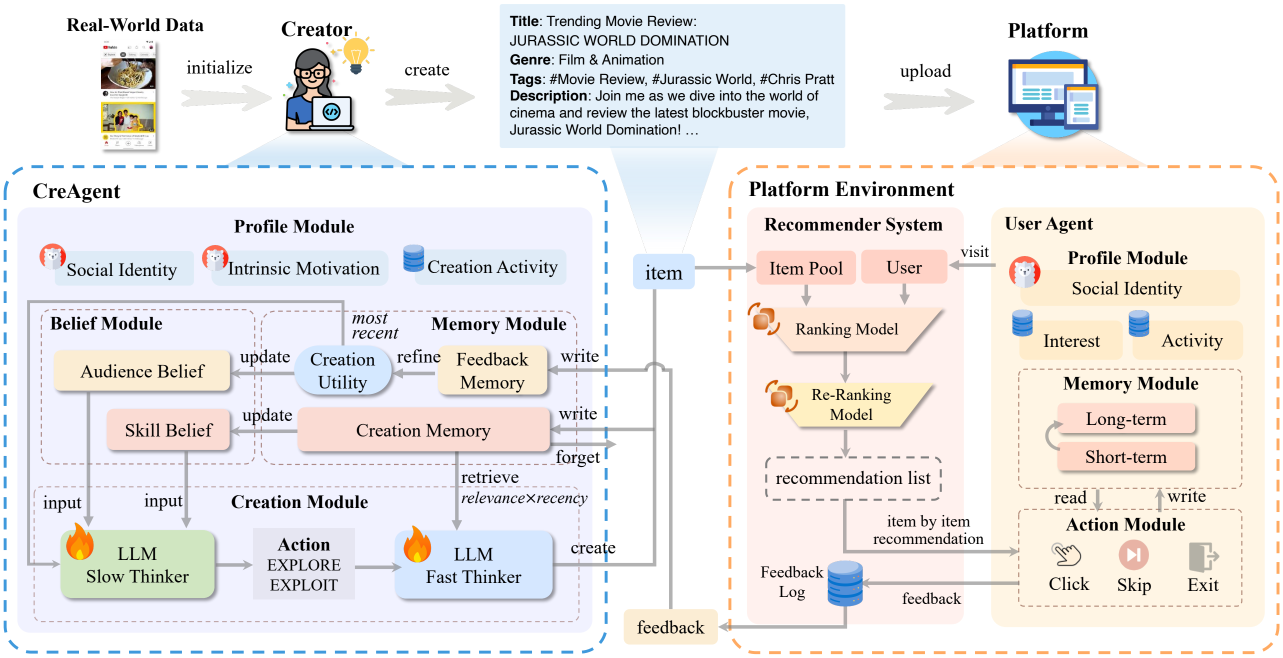
4.1 Profile Module Design

In this module, we initialize CreAgent's profile using a pre-collected real-world dataset, crucial for aligning agent behavior with real human actions. CreAgent's profile comprises three elements: social identity, intrinsic motivation, and creation activity.

Social identity. Content creation, as a form of social behavior, is driven by not only economic interests but also social identity, a factor often overlooked in behavior modeling [61, 62, 66]. To bridge this gap, we employ LLM to identify creators' social identities by analyzing their creation history $\mathcal{I}_c^0$ and basic information. For example, a creator who enjoys producing lifestyle content might be identified as a "lifestyle advocate", while one who frequently creates gaming content could be summarized as a "gaming enthusiast". We represent this social identity as P_c^s, whose format is in text.

Intrinsic motivation. Established studies [5, 15] show that intrinsic motivation is a key factor that affects creators' creation behavior, yet this motivation varies from person to person. For example, while some creators frequently produce content to obtain more revenue, others create simply to share their lives without high-profit expectations. We use LLM to help us summarize each creator's intrinsic motivation P_c^m from their creation history $\mathcal{I}_c^0$ and frequency as one part of their profiles.

Creation activity. In addition to the above two economic characteristics, creation activity (i.e., the frequency of a creator's creation) is also an important trait of the creator. We use the collected dataset $\mathcal{I}_c^0$ to initialize the inherent activity level p_c^a for each creator c, defined as the average number of items created per day by that creator. To normalize creation frequency across creators, we define the probability that creator c generates content in each period as p_c^a/η, where $\eta = \max\{p^a\}$ denotes the highest activity level among all creators.



(a) CreAgent (b) Platform environment

Figure 2: The overall workflow of our simulation platform. (a) CreAgent, initialized with the real-world YouTube dataset, employs a belief mechanism combined with fast-and-slow thinking to create the next item strategically based on platform-provided user feedback. (b) Platform environment, which consists of an extensible two-stage recommender system and modified widely-used user agents, collects item feedback from users and sends it to CreAgent.

4.2 Memory Module Design

Considering the two key pieces of information that creators prioritize and store in reality, we build two memories for CreAgent: feedback memory and creation memory.

Feedback memory. Different from user agent memory modeling [53], the user feedback received by each creator is usually recorded and maintained by the platform over a considerable period [26]. At the end of each time period t, the feedback memory $\mathcal{M}_c^{\text{feed}}$ of creator c will be updated based on the user feedback received for the creator's historical items $\mathcal{I}_c$:

$$\mathcal{M}_c^{\text{feed}} = \mathcal{M}_c^{\text{feed}} \cup \left\{ \sum_{u\in\mathcal{U}^t} \mathrm{E}_{u,i}(t), \sum_{u\in\mathcal{U}^t} \mathrm{Y}_{u,i}(t) \mid i \in \mathcal{I}_c \right\}. \quad (1)$$

After storing user feedback in memory $\mathcal{M}_c^{\text{feed}}$ at each time period t, a refinement process is applied to compute the utility of each created item. Specifically, the utility is calculated based on the time-averaged exposure and click counts for each item $i \in \mathcal{I}_c$ up to the current period t.

$$\mathbf{z}_i(t) = \frac{1}{\theta}\left[\beta \sum_{n=t(i)}^{t}\sum_{u\in\mathcal{U}^k} \mathrm{E}_{u,i}(n) + (1-\beta)\sum_{n=t(i)}^{t}\sum_{u\in\mathcal{U}^k}\mathrm{Y}_{u,i}(n)\right], \quad (2)$$

where $\theta = \frac{1}{t-t(i)+1}$ and $t(i)$ is the creation time period of item $i \in \mathcal{I}_c$, β is the hyperparameter that controls the importance of exposure for the creator. This utility can be configured by adjusting β according to specific scenarios, either as exposure-based [39, 59] or click-based [63]. The utility $\mathbf{z}$ will be used as input for both the belief module (Section 4.3) and the creation module (Section 4.4) to help CreAgent make future creation decisions.

Creation memory. The creation memory $\mathcal{M}_c^{\text{cre}}$ stores the historical creation item information of the creator agent c. To better reflect the human time-fading memory mechanism, we draw on prior studies and incorporate a power function forgetting rate [53]. Before each creation, creators will retrieve the most relevant and recent creation experience from the creation memory. After each creation at time period t, the content of the newly created item will be stored in the creator's creation memory: $\mathcal{M}_c^{\text{cre}} = \mathcal{M}_c^{\text{cre}} \cup \{i \mid i \in \mathcal{I}_c \cap \mathcal{I}^n\}$. Then, the creation memory $\mathcal{M}_c^{\text{cre}}$ will be used to update the skill belief (Section 4.3) and as the input of fast thinker (Section 4.4) for future creation.

4.3 Belief Module Design

Under information asymmetry conditions, creators usually do not have exact information about other stakeholders (e.g., user preferences and item genres), and their creation behavior is often driven by their limited information expectations about it (i.e., beliefs). In this module, inspired by game theory [40], we incorporate a belief mechanism to model the limited information setting and guide the strategic content creation of CreAgent. The beliefs of CreAgent are typically formed based on previous information provided by the platform and are updated over time and with the acquisition of new information. Specifically, the strategic creation behavior of

CreAgent is mainly driven by two types of beliefs: skill belief and audience belief.

Skill belief. Under limited information, creators have incomplete knowledge of item genres. They will gradually acquire more genre information and improve their creation proficiency in those genres during the creative process. The skill belief represents the creator's confidence in their ability to create each genre of items, which is defined as the percentage of their content created in each genre. At the start of each time period t, the skill belief of creator c for genre g will be updated based on the creation memory $\mathcal{M}_c^{\text{cre}}$:

$$\mathbf{B}_{c,g}^{\text{skill}}(t) = \frac{|\mathcal{I}_g \cap \mathcal{I}_c|}{|\mathcal{I}_c|}, \quad \forall g \in \mathcal{G}, \mathcal{I}_g, \mathcal{I}_c \subset \mathcal{M}_c^{\text{cre}}. \tag{3}$$

Due to some creators being more adept at producing certain types of content than others (e.g., due to specific talent, interests, and facilities), we initialize the skill belief $\mathbf{B}_c^{\text{skill}}(1)$ using the pre-collected dataset $\mathcal{I}_c^0$.

Audience belief. The audience belief represents the creator's internal understanding and expectation of user preferences in each genre. At the start of each time period t, the audience belief of creator c in genre g will be updated based on the user feedback stored in the feedback memory $\mathcal{M}_c^{\text{feed}}$:

$$\mathbf{B}_{c,g}^{\text{aud}}(t) = \frac{\sum_{i \in \mathcal{I}_g \cap \mathcal{I}_c} \mathbf{z}_i(t)}{|\mathcal{I}_g \cap \mathcal{I}_c|}, \quad \forall i \in \mathcal{I}_c. \tag{4}$$

Similarly to the skill belief, we initialize the audience belief $\mathbf{B}_c^{\text{aud}}(1)$ using the real-world creator's history items $\mathcal{I}_c^0$.

4.4 Creation Module Design

In this module, we apply the fast-and-slow thinking mechanism [20, 28, 64] to equip our creator agents with human-like analysis and creation abilities. The thought process is composed of two phases [20]: slow thinking for strategic planning and analysis, and fast thinking for rapid content generation based on experience and instinct.

Slow thinker. During each creation, user feedback on the most recently created item directly affects the creator's judgment on whether to continue with the current creative strategy or change it. At the start of each time period t, creator $c \in C^t$ gets three key factors that affect his creation strategy as inputs: (1) the utility of the most recently created item i, i.e.,$\mathbf{z}_i(t)$, (2) the skill belief $\mathbf{B}_c^{\text{skill}}$ and audience belief $\mathbf{B}_c^{\text{aud}}$, and (3) the social identity P_c^s and intrinsic motivation P_c^m. Then, these inputs are then fed into the LLM via the designed Chain-of-Thought [56] prompt P_1 for slow thinking:

$$\mathbf{A}_c^{\text{exp}}(t) = \text{LLM}\left[P_1\left(g(i), \mathbf{z}_i(t), \mathbf{B}_c^{\text{aud}}(t), \mathbf{B}_c^{\text{skill}}(t), P_c^s, P_c^m\right)\right], \tag{5}$$

where $i = h(\mathcal{I}_c)$ represent the most recent item created by creator c, where $h(\cdot)$ fetches the new item within $(\cdot)$, and $g(i)$ denotes the genre of item i. $\mathbf{A}_c^{\text{exp}}(t)$ is the text-based explore/exploit action taken by the creator agent c at time period t. Specifically, CreAgent will choose whether to continue creating within the current genre $g(i)$ or to explore another genre $\mathcal{I}/\{g(i)\}$.

Fast thinker. After generating analytical results, the creator agent will produce content based on these findings. The content is primarily divided into four sections: item title, item genre, item tags, and item description. Before creating content, the creator agent will retrieve the creation experience $f(\mathcal{M}_c^{\text{cre}})$ from the creation memory $\mathcal{M}_c^{\text{cre}}$ based on the action $\mathbf{A}_c^{\text{exp}}(t)$, to assist the fast thinker in the creation process.

$$\mathbf{A}_c^{\text{cont}}(t) = \text{LLM}\left[P_2\left(f\left(\mathcal{M}_c^{\text{cre}}, \mathbf{A}_c^{\text{exp}}(t)\right), P_c^s, P_c^m\right)\right], \tag{6}$$

where P_2 is the designed prompt for fast thinking. $f(\cdot_1, \cdot_2)$ is the retrieval function which retrieves the most recent creation memory of genre $(\cdot_2)$ from memory bank $(\cdot_1)$. Intuitively, CreAgent generates the content of the next item, $\mathbf{A}_c^{\text{cont}}(t)$—including the title, genre, tags, and description—based on its creation memory and creator profile.

4.5 Fine-tuning to Enhance Creative Ability

In this section, we use the Proximal Policy Optimization (PPO) algorithm [43] to fine-tune CreAgent. By simulating the real-world creation cycle of creating, receiving rewards, analyzing, and creating again, we aim to: (1) improve CreAgent's understanding of creators' limited information status and user feedback; (2) enhance CreAgent's analytical and creative capabilities.

Reward modeling. In real platform scenarios, creators receive user feedback on their created items from the platform and benefit from it, which also helps them adjust their existing creation strategies. In this paper, we utilize the PPO algorithm and treat the the platform environment (Section 5) as the reward model, to replicate the process through which creators learn their creative strategies in the real world. For each created item $i \in \mathcal{I}_c$, the reward is the weighted utility until the current time period t, i.e., $\lambda \mathbf{z}_i(t)$, where λ is a constant coefficient used to ensure training stability by preventing the reward from being too large or too small.

Replay buffer. In real-world scenarios, user feedback is often not collected immediately but takes some time to accumulate. After the creation of item i at time period $t = t(i)$, we first store the state $s_n^c = (\mathbf{B}_{c,g}^{\text{skill}}(t), \mathbf{B}_{c,g}^{\text{aud}}(t)))$ and action $a_t^c = (\mathbf{A}_c^{\text{exp}}(t), \mathbf{A}_c^{\text{cont}}(t))$ in the log. After accumulating T_r periods until period $t + T_r$, we save the reward $r_n^c = \lambda \mathbf{z}_i(t + T_r)$, along with the corresponding state s_n^c, and action a_t^c, into the replay buffer as a tuple (s_t^c, a_t^c, r_t^c). For every T_u period, we update the policy by sampling M records from the replay buffer.

PPO Fine-tuning. To avoid the LLM policy π_θ with parameter θ deviating from the initial reference policy $\pi_{\theta_{\text{init}}}$ too far, we follow [38] and introduce the KL-divergence penalty into the current reward function. Therefore, the policy optimization formula is:

$$\mathcal{L}_{PPO} = \mathbb{E}\left[\lambda \mathbf{z}_i(t) - \beta KL\left(\pi_{\theta_{\text{init}}}, \pi_\theta\right)\right], \tag{7}$$

where λ is the hyper-parameter that prevents the final reward $\mathbf{z}_i(t)$ from becoming excessively large to ensure stable fine-tuning.

5 Platform Environment

In this section, we introduce a simulated platform environment that mirrors the one creators engage with. This environment is primarily for receiving items uploaded by creators and providing them with user feedback. As illustrated in Figure 2 (b), the platform environment is mainly divided into the RS and user agents.

Recommender system modeling. The RS of the simulation platform primarily focuses on extensibility. As shown in Figure 2, it encompasses a dynamic item set that allows items to be added freely, a user set, a two-stage recommendation process that includes ranking and re-ranking, and a feedback log to store user feedback. We discuss two aspects of environment construction that resonate with real-world RS, including two-stage ranking and item-by-item recommendation. • **Two-stage ranking.** In real industrial scenarios, recommendation lists are often generated through a multi-stage ranking process [30]. We also consider this situation by allowing both the ranking and re-ranking algorithms to be flexibly replaced according to specific requirements. Through this design, the platform can flexibly adjust and evaluate different recommendation strategies, whether by applying only the ranking model (e.g., BPR [41]) or by considering long-term objectives (e.g., fairness [54] and diversity [25]) in the re-ranking stage. • **Item-by-item recommendation.** After generating the recommendation list, RS will recommend items to users on an item-by-item basis. This scenario is currently widely used in online recommendation platforms such as YouTube [47] and TikTok.

User agents. Our simulation platform is compatible with any reasonable user simulator for RS [46, 53, 67]. In the experiment presented in this paper, we employ the widely used LLM-based user simulator, RecAgent [53], as our user agent. By integrating sensory, short-term, and long-term memory, RecAgent effectively simulates the human memory mechanism, capturing both short-term and long-term user interests. Also, to better reflect users' long-term behavior, we make some targeted changes. For the profile, we summarize a targeted social identity for each user by LLM based on their interaction and comment history. The user's long-term interests and activity levels are also derived from pre-collected real-world dataset $\mathcal{U}^0$. After being recommended, the user agent can take three actions: click the item, skip the item, or exit the RS. For details on the alignment evaluation of the user agent, please refer to the GitHub repository[1].

6 Creator Credibility Evaluation

In this section, we conduct experiments to verify the effectiveness of CreAgent in long-term simulation by addressing the following research questions:

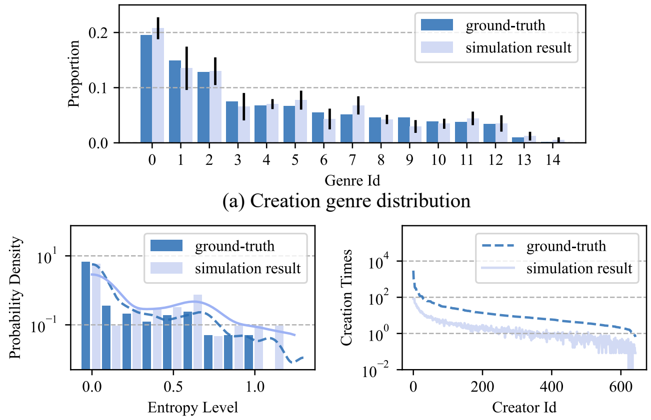
RQ1: Can CreAgent well align with the real-world creator patterns?
RQ2: Can CreAgent well simulate the creation behavior under information asymmetry conditions in the real world?
RQ3: What is the computational cost of our simulation platform?
CreAgent and the simulation platform are available at GitHub[1].

6.1 Experimental Setups

Real-world dataset collection. We initiate a focused data collection effort to tackle the lack of recommendation datasets that include detailed information about content creators and their items. We leverage the YouTube Data API to collect item information and comment data from channels in *YouTube* [47], the world's leading content platform, known for its diverse and influential content creators. This choice is strategic to ensure our simulation closely mirrors the real-world platform. Specific details of the dataset are provided in the GitHub repository[1].

Simulation setups. We utilize two A6000 GPUs for simulations, setting the number of user agents $|\mathcal{U}| = 100$ and creator agents

[1]https://github.com/shawnye2000/CreAgent.git



(a) Creation genre distribution

(b) Creation diversity

(c) Creation activity

Figure 3: Comparison between the creation genre preference, diversity, and activity of the ground-truth and agent-simulated result.

$|C| = 50$, with a total of $T = 100$ simulation periods. All agents are powered by the Llama3-8B [52], updated and executed in parallel using multi-threading. The recommendation list length $K = 5$. The RS model is trained on the real-world dataset $\mathcal{D}^0$ and retrained every $T_{\text{train}} = 5$ period using all interaction data $\mathcal{D}^0 \cup \mathcal{D}$ to reflect the periodic update of real-world RS. This section uses the Deep Interest Network (DIN) [71] as the base RS model for the credibility evaluation. To better align with real-world content platform scenarios [27, 68] and help to cold-start new items, we consider the timeliness of item recommendation by removing excessively outdated items from the item pool.

Baselines. We select several classic creator behavior simulation baselines, where the baseline methods determine the genre of the next created item during each time period. Creator Feature Dynamics (**CFD**) [29, 66]: creators adjust their creation strategy using user feedback as the gradient, scaled by a learning rate. Local Better Response (**LBR**) [63]: creators generate a random direction and evaluate its utility (impact on user feedback). If beneficial, they update their strategy incrementally; otherwise, they maintain the current strategy. **SimuLine** [69]: creators' next creation is determined through probabilistic sampling based on the number of likes from previous periods.

6.2 RQ1: Real-world Dataset Alignment

We first assess the alignment between simulated and real-world creator patterns across preferences, diversity, and activity. Then, an ablation study highlights the contribution of key modules. Content-level alignment results are available on GitHub[1] due to space limits.

6.2.1 ***Creation Preference Alignment.*** On real platforms, creators exhibit diverse preferences that influence their content creation. To faithfully replicate this in simulation, we first ensure that the simulated preference distribution aligns with that observed in real-world data.

As shown in Figure 3 (a), we plot the creation genre distribution of the YouTube dataset (i.e., the light blue bar) and our simulation

Table 1: Comparison of the divergence between the simulated and real-world distributions using Jensen-Shannon divergence [31], with genre-level for preference evaluation and individual-level for diversity.

Creator Modeling Method	Preference	Diversity
Creator Feature Dynamics [29, 66]	0.2537	0.7204
Local Better Response [63]	0.2833	0.6284
SimuLine [69]	0.3175	0.6949
CreAgent (w/o Audience Belief)	0.2671	0.6159
CreAgent (w/o Skill Belief)	0.2928	0.5532
CreAgent (w/o Fast-Slow Thinker)	0.1728	0.5638
CreAgent	**0.1667**	**0.3014**

(i.e., the deep blue bar). Specifically, the x-axis represents item genres, and the y-axis is the proportion of creation times all CreAgents created in each genre during the first 10 environment initialization periods. From the comparison, we observe that our creator agents ultimately achieved a genre distribution comparable to the real-world dataset, effectively replicating a similar content ecosystem. Additionally, we notice some differences in certain categories (e.g., CreAgents show a stronger preference for genre 7 and less for genre 6 compared to real creators), which we attribute to the influence of the LLM's pre-trained knowledge. Table 1 demonstrates that CreAgent exhibits greater consistency with the dataset's preference distribution compared to the baselines.

6.2.2 ***Creation Diversity Alignment.*** Creation diversity varies among creators: some specialize in a single category, while others engage across multiple ones. Figure 3 (b) presents a histogram of the creation diversity for creators on the simulated platform and those in the YouTube dataset. We use the entropy of genre frequencies to represent each creator's diversity (the x-axis). The solid lines in the figure represent the Kernel density estimation (KDE) curves [55]. We observe that CreAgent effectively replicate the diversity distribution observed in the YouTube dataset: most creators stick to a single genre, while a few actively explore different genres (the y-axis is on a log scale). The superiority compared to the baseline is demonstrated in Table 1.

6.2.3 ***Creation Activity Alignment.*** To reflect real-world variation in creator activity, we assess how well CreAgent aligns with observed creation frequencies. Figure 3(c) visualizes the distribution at the individual level. The deep blue dashed line represents the average creation count per month for 643 creators in the YouTube dataset (in decreasing order), while the light blue area represents the corresponding CreAgent's total creation times under 100 periods' simulation. Since we sample 50 creators per simulation, we conduct 20 experiments to count the average creation times of all 643 creators. We can observe that the simulation results are very consistent with the actual distribution of creation counts, both exhibiting a long-tail distribution (the y-axis is on a log scale).

6.2.4 ***Ablation Study.*** We conduct an ablation study to investigate the impact of several key modules of CreAgent on creator behavior simulation and alignment (i.e., audience/skill belief, fast-slow thinker), as shown in Table 1. We find that both audience belief and skill belief have a significant impact on preference alignment, as they contain information about real creators. Regarding diversity alignment, removing any of these three components greatly affects the results, proving that CreAgent requires all of them to achieve alignment with real-world creative diversity.

6.3 RQ2: Interaction Behavior Alignment

Under information asymmetry, the long-term interaction behaviors of creators with the platform (i.e., creation behavior) follow certain patterns, which is supported by renowned behavioral economics theory [35]. To evaluate such interaction behavior of CreAgent, we selected the two most fundamental theories from behavioral economics: bounded rational [9, 44] and prospect theory [21].

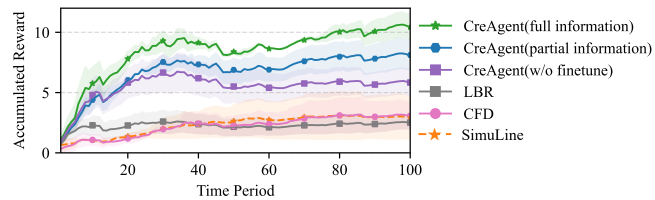


Figure 4: Accumulated reward of creator agents over time periods, normalized by random creation strategy [45].

6.3.1 ***Bounded Rationality.*** Established behavioral economic studies [9, 44] have revealed that *individuals may not always be fully rational when making decisions. Their choices can be influenced by limited information, leading to suboptimal decisions.*

Figure 4 shows the normalized cumulative rewards [45] of CreAgent under full and partial information scenarios compared with other baselines. Full information refers to providing CreAgent with extra information regarding the distribution of user preferences in each genre. We can observe that CreAgent under full information achieves a higher reward than CreAgent under partial information, suggesting that CreAgent under limited information cannot always make fully rational decisions to maximize its reward. Limited information leads CreAgent to make suboptimal choices, which is consistent with human behavior under limited information [9]. Additionally, when comparing CreAgent with the baselines, our approach demonstrates a higher accumulated reward, which indicates that CreAgent leverages human-like analyzing and creation abilities of LLM, exhibiting stronger decision-making and creative capabilities under limited information conditions.

6.3.2 ***Prospect Theory.*** With only limited information available, the user feedback of creators' creation action is uncertain and at risk, presenting them with the dilemma of exploration/exploitation. They can either continue creating content they are familiar with (i.e., exploitation) or try unfamiliar fields (i.e., exploration). The well-known behavior economic study Prospect Theory [21, 22] suggests that: *individuals often exhibit asymmetric behavior in decision-making under uncertainty, where they are more sensitive to losses than to equivalent gains, leading to risk-avoiding in high-reward situations and risk-embracing in low-reward situations.*

As reported in Figure 5, we plot the creator agent's next action proportion at different reward levels on the last created item. In

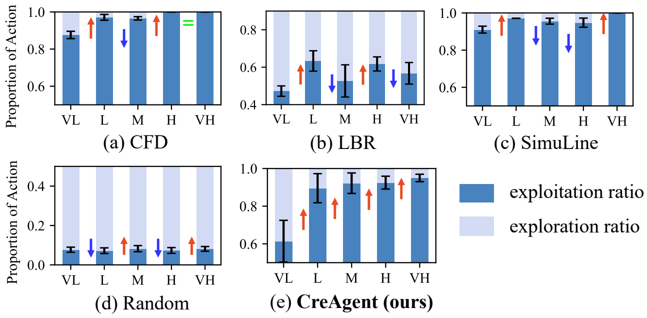
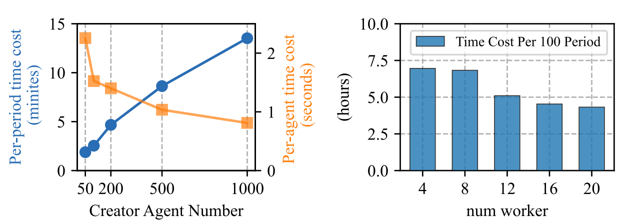


Figure 5: Agent's next action proportion under different reward levels, with VL, L, M, H, and VH for very low, low, medium, high, and very high, respectively. The exploitation ratio of a higher reward level can be higher ↑, equal =, or lower ↓ to that of a lower reward level.

Figure 5(e), when CreAgents under limited information receive a relatively low reward for their previous creation, their creation strategies are more aggressive (i.e., exploration action is nearly 40%). Conversely, if they receive a high reward, they tend to stick to their existing content strategy to avoid risk, as evidenced by increased exploitation proportion from low to high rewards. For comparison, as shown in Figure 5 (a-d), traditional simulation baselines cannot align with such prospect theory principles, showing a non-monotonic trend in exploration-exploitation ratio as the previous creation reward gradually increases.

6.4 RQ3: Computaional Costs

Though currently limited to small-scale simulations, we discuss computational complexity and costs to show that our simulator can support large-scale simulations with sufficient resources. As shown in Figure 6 (a), as the number of simulated CreAgents increases, the time cost per period does not rise to an unacceptable level. Even with 1000 agents, the time cost per period stays under 15 minutes, offering a significant computational cost advantage over online A/B testing, which usually takes weeks or months for long-term evaluations. Moreover, we observe that the average computational cost per agent decreases as the number of agents increases. Additionally, Figure 6 (b) shows that increasing the number of CPU workers can further reduce the time costs through multi-threaded parallelism, enabling large-scale simulations.



(a) Time cost w.r.t. agent number (b) Time cost w.r.t. num worker

Figure 6: (a) Time cost per agent and time cost per period w.r.t. the number of simulated agents; (b) Total simulation time for 100 periods w.r.t. the number of CPU workers.

7 Long-term Effect Evaluation

In this section, we use CreAgent and the simulation platform to assess the long-term effect of RS on different stakeholders. Our evaluation focuses on three key long-term objectives that the multi-stakeholder content platform concerns [11, 53, 59]: Can different RS models (1) improve user engagement (**RQ4**), (2) protect content creators (**RQ5**), and (3) enrich content (**RQ6**) in the long run?

7.1 Experimental Setups

Evaluation setups. The experimental setup in this section is identical to Section 6. Note that the first $T_0 - 1 = 9$ periods are for environment initialization, followed by 90 evaluation periods starting at $T_0 = 10$. In this paper, we consider the practical issue of creator retention, assuming a creator leaves if no clicks are received after 5 consecutive creations [32, 59].

RS model selection. In this paper, we mainly focus on evaluating three types of RS models. (1) Basic ranking models: Random, most popular (Pop), MF [24], BPR [41], DIN [71]; (2) Diversity-aware models: MMR [7], APDR [51]; (3) Provider (creator) fairness-aware models: P-MMF [59], FairRec [39], CPFair [36], TFROM [58], FairCo [34]. For MF and BPR, we use the user and item IDs as input features. For DIN, we incorporate the embedding of the generated textual content, along with the creator ID, genre ID, and the user's historical interaction sequence.

Table 2: Rankings model evaluations, with five runs per model using various seeds, display mean and standard deviation with larger and smaller numbers, respectively.

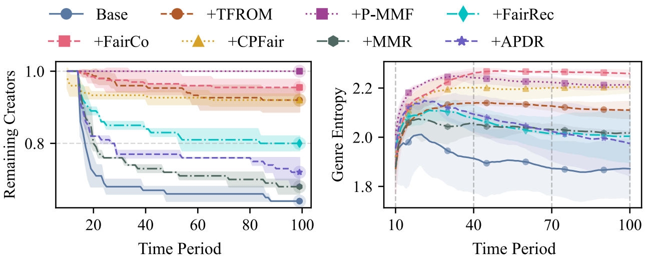
Models	User Welfare	Creator Retention	Content Diversity
Random	$6220_{\pm 159.8}$	$\mathbf{0.900}_{\pm 0.000}$	$2.221_{\pm 0.045}$
Pop	$6167_{\pm 647.0}$	$0.500_{\pm 0.028}$	$2.146_{\pm 0.095}$
MF	$9322_{\pm 104.7}$	$0.710_{\pm 0.014}$	$2.190_{\pm 0.035}$
BPR	$8554_{\pm 353.4}$	$0.620_{\pm 0.000}$	$\mathbf{2.223}_{\pm 0.101}$
DIN	$\mathbf{11289}_{\pm 1353}$	$0.627_{\pm 0.012}$	$1.872_{\pm 0.145}$

Table 3: Evaluation of fairness- and diversity-aware models with DIN as the base model. Setups are same as Table 2.

Models	User Welfare	Creator Retention	Content Diversity
Base	$11289_{\pm 1353}$	$0.627_{\pm 0.012}$	$1.872_{\pm 0.145}$
Diversity-aware			
+MMR	$11059_{\pm 114.6}$	$0.680_{\pm 0.028}$	$2.017_{\pm 0.008}$
+APDR	$13489_{\pm 215.7}$	$0.720_{\pm 0.057}$	$1.974_{\pm 0.185}$
Fairness-aware			
+FairRec	$14108_{\pm 530.1}$	$0.840_{\pm 0.028}$	$1.918_{\pm 0.133}$
+FairCo	$12749_{\pm 1955}$	$0.960_{\pm 0.000}$	$\mathbf{2.246}_{\pm 0.018}$
+TFROM	$11089_{\pm 326.7}$	$0.920_{\pm 0.028}$	$2.144_{\pm 0.048}$
+P-MMF	$13865_{\pm 225.6}$	$\mathbf{1.000}_{\pm 0.000}$	$2.228_{\pm 0.020}$
+CPFair	$\mathbf{14506}_{\pm 605.3}$	$0.940_{\pm 0.028}$	$2.186_{\pm 0.002}$

7.2 RQ4: User Engagement

Motivation. Long-term user engagement has always been one of the key goals pursued by the platform [2, 11]. We present the Total



(a) Remaining creator proportion (b) Long-term genre diversity

Figure 7: Changes in (a) remaining creators and (b) content diversity over time under different re-ranking models.

User Welfare (TUW) metric for evaluation, which is defined as the cumulative user click number:

$$\text{Total User Welfare} = \sum_{t=T_0}^{T} \sum_{u \in \mathcal{U}^t} \sum_{i \in \mathcal{I}^t} \mathrm{Y}_{u,i}(t). \tag{8}$$

Results. Table 2 shows the long-term user engagement for various ranking models. Naive strategies like Random and Pop perform poorly in this context. ID-based models like MF and BPR show moderate improvements but still struggle to fully capture user interests. In contrast, feature-based models such as DIN more effectively model long-term user preferences, resulting in significantly improved user engagement.

Table 3 indicates that diversity- and fairness-aware models also maintain strong long-term user engagement. For instance, MMR reduces engagement by only 2.03%, and TFROM by 1.77% compared to the base model. Notably, some fairness-aware models like CPFair even significantly boost engagement, achieving a 28.50% increase. This phenomenon can be attributed to the exploration of users' latent interests. Unlike short-sighted strategies (e.g., Pop) tend to overemphasize users' current preferences while overlooking their latent interests, these models are more effective in uncovering these potential interests, thereby enhancing long-term engagement.

7.3 RQ5: Creator Protection

Motivation. Creator protection is crucial for platform growth and user attraction [2]. However, short-sighted recommendation strategies may prevent niche creators from gaining enough exposure, potentially leading to their departure from the platform [14, 58, 59] and reduce content diversity. To assess the protection of creators of different recommendation models (**RQ5**), we propose the Creator Retention Rate (CRR) metric, which is defined as the alive creator (i.e., creators who have not yet left) number in the T-th period divided by the alive creator number in the T_0-th period.

Result. Table 2 and Table 3 show the creator protection performance of different RS models. Aside from the random strategy's high CRR (which gives equal exposure to each item), other strategies have low CRR (below 0.8), suggesting a loss of over 20% of initial creators after 100 simulation periods.

Figure 7 (a) illustrates the remaining creator proportion as the simulation progresses. The base model (DIN) experiences a rapid decline in creators, whereas diversity-aware and fairness-aware algorithms significantly slow this departure. Notably, different fairness algorithms achieve varying degrees of creator protection. For instance, P-MMF [59], designed to boost exposure for worst-exposed creators, retains all creators by the 100th time period, thus effectively safeguarding the platform's creator ecosystem.

7.4 RQ6: Content Enrichment

Motivation. The enrichment of content is a critical issue for online platforms. Myopic strategies may lead to falling into a filter bubble effect [37, 53], causing decreased user experience, creator exposure unfairness [54], and insufficient exploration of user interests [8]. To measure the level of content enrichment, we follow [53] and define Content Genre Diversity (CGD) as the average genre entropy of items received by each user. It can be defined as follows:

$$\text{Content Diverisity} = -\frac{1}{\sum_{t=T_0}^{T} |\mathcal{U}^t|} \sum_{t=T_0}^{T} \sum_{u \in \mathcal{U}^t} \sum_{g \in \mathcal{G}} p_{u,g} \log\left(p_{u,g}\right), \tag{9}$$

where $p_{u,g} = \frac{\sum_{t=T_0}^{T} \sum_{u \in \mathcal{U}^t} \sum_{i \in I_g} \mathrm{E}_{u,i}(t)}{\sum_{t=T_0}^{T} \sum_{u \in \mathcal{U}^t} \sum_{i} \mathrm{E}_{u,i}(t)}$ is the frequency of genre g recommended to user u. A higher CGD value indicates greater content enrichment and a milder filter bubble effect.

Results. As reported in Table 2, despite the high long-term user engagement DIN achieves (mentioned in Section 7.2), it goes through a pronounced filter bubble effect, with a CGD value 15.8% lower than BPR. We attribute this to its reliance on genre and creator features, which leads to greedy recommendations on the known interests of users. We also examine the long-term impacts of diversity-aware and fairness-aware models on enriching the content, as illustrated in Figure 7 (b) and Table 3. Compared with the base model whose genre diversity (CGD) declines over time, diversity-aware strategies alleviate this, and fairness-aware strategies notably increase recommendation diversity, as evidenced by FairCo maintaining a consistently high enrichment over time.

8 Conclusion and Future Work

In this paper, we propose an LLM-empowered creator simulator to enhance the long-term evaluation of RS under information asymmetry scenarios. Extensive experiments demonstrate CreAgent's effectiveness in aligning with real-world creator behavior and providing valuable insights into the long-term effects of various RS models in the multi-stakeholder environment.

Due to the complexity and randomness of real-world content creation behavior, our system is built upon several simplified assumptions, which are often inevitable in simulation-based studies. The main assumptions include: (1) creators adjust their new content creation mainly based on user feedback received on their previously created items; (2) creators can only produce text-based items. We leave the relaxation of these assumptions to future work, aiming to bridge the gap between simulations and real-world dynamics.

Acknowledgments

This work was funded by the National Key R&D Program of China (2023YFA1008704), the National Natural Science Foundation of China (62472426), Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, funds for building world-class universities (disciplines) of Renmin University of China.

References

[1] Himan Abdollahpouri and Robin Burke. 2019. Multi-stakeholder recommendation and its connection to multi-sided fairness. *arXiv preprint arXiv:1907.13158* (2019).

[2] Amit Kumar Bardhan and Saad Ashraf. 2022. More buyers or more sellers: on marketing resource allocation strategies of competing two-sided platforms. *Electronic Commerce Research* (2022), 1–30.

[3] Omer Ben-Porat and Moshe Tennenholtz. 2018. A game-theoretic approach to recommendation systems with strategic content providers. *Advances in Neural Information Processing Systems* 31 (2018).

[4] Hemant K Bhargava. 2022. The creator economy: Managing ecosystem supply, revenue sharing, and platform design. *Management Science* 68, 7 (2022), 5233–5251.

[5] Xiang Bi and Cunchen Tang. 2020. Research on the motives affecting the behavior of short video's creators. *Ieee Access* 8 (2020), 188415–188428.

[6] Robin D Burke, Himan Abdollahpouri, Bamshad Mobasher, and Trinadh Gupta. 2016. Towards multi-stakeholder utility evaluation of recommender systems. *UMAP (Extended Proceedings)* 750 (2016).

[7] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*. 335–336.

[8] Minmin Chen, Yuyan Wang, Can Xu, Ya Le, Mohit Sharma, Lee Richardson, Su-Lin Wu, and Ed Chi. 2021. Values of user exploration in recommender systems. In *Proceedings of the 15th ACM Conference on Recommender Systems*. 85–95.

[9] John Conlisk. 1996. Why bounded rationality? *Journal of economic literature* 34, 2 (1996), 669–700.

[10] Yujuan Ding, Yunshan Ma, Wenqi Fan, Yige Yao, Tat-Seng Chua, and Qing Li. 2024. Fashionregen: Llm-empowered fashion report generation. In *Companion Proceedings of the ACM on Web Conference 2024*. 991–994.

[11] Pejman Ebrahimi, Datis Khajeheian, Maryam Soleimani, Abbas Gholampour, and Maria Fekete-Farkas. 2023. User engagement in social network platforms: what key strategic factors determine online consumer purchase behaviour? *Economic Research-Ekonomska Istraživanja* 36, 1 (2023), 2106264.

[12] Lana El Sanyoura and Ashton Anderson. 2022. Quantifying the creator economy: A large-scale analysis of patreon. In *Proceedings of the International AAAI Conference on Web and Social Media*, Vol. 16. 829–840.

[13] Adrienne Felt and David Evans. 2008. Privacy protection for social networking platforms. Citeseer.

[14] Henning Hohnhold, Deirdre O'Brien, and Diane Tang. 2015. Focusing on the long-term: It's good for users and business. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1849–1858.

[15] Clark Leonard Hull. 1943. Principles of behavior: an introduction to behavior theory. (1943).

[16] Eugene Ie, Chih-wei Hsu, Martin Mladenov, Vihan Jain, Sanmit Narvekar, Jing Wang, Rui Wu, and Craig Boutilier. 2019. Recsim: A configurable simulation platform for recommender systems. *arXiv preprint arXiv:1909.04847* (2019).

[17] Dietmar Jannach and Christine Bauer. 2020. Escaping the McNamara fallacy: Towards more impactful recommender systems research. *Ai Magazine* 41, 4 (2020), 79–95.

[18] Dietmar Jannach, Markus Zanker, Mouzhi Ge, and Marian Gröning. 2012. Recommender systems in computer science and information systems–a landscape of research. In *E-Commerce and Web Technologies: 13th International Conference, EC-Web 2012, Vienna, Austria, September 4-5, 2012. Proceedings 13*. Springer, 76–87.

[19] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)* 20, 4 (2002), 422–446.

[20] Daniel Kahneman. 2011. Thinking, fast and slow. *Farrar, Straus and Giroux* (2011).

[21] Daniel Kahneman and Amos Tversky. 2013. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*. World Scientific, 99–127.

[22] Daniel Kahneman and Amos Tversky. 2013. Prospect theory: An analysis of decision under risk. In *Handbook of the fundamentals of financial decision making: Part I*. World Scientific, 99–127.

[23] Ron Kohavi and Roger Longbotham. 2015. Online controlled experiments and A/B tests. *Encyclopedia of machine learning and data mining* (2015), 1–11.

[24] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37.

[25] Matevž Kunaver and Tomaž Požrl. 2017. Diversity in recommender systems–A survey. *Knowledge-based systems* 123 (2017), 154–162.

[26] Patricia G Lange. 2007. Publicly private and privately public: Social networking on YouTube. *Journal of computer-mediated communication* 13, 1 (2007), 361–380.

[27] Fengqi Liang, Baigong Zheng, Liqin Zhao, Guorui Zhou, Qian Wang, and Yanan Niu. 2024. Ensure Timeliness and Accuracy: A Novel Sliding Window Data Stream Paradigm for Live Streaming Recommendation. *arXiv preprint arXiv:2402.14399* (2024).

[28] Bill Yuchen Lin, Yicheng Fu, Karina Yang, Faeze Brahman, Shiyu Huang, Chandra Bhagavatula, Prithviraj Ammanabrolu, Yejin Choi, and Xiang Ren. 2024. Swiftsage: A generative agent with fast and slow thinking for complex interactive tasks. *Advances in Neural Information Processing Systems* 36 (2024).

[29] Tao Lin, Kun Jin, Andrew Estornell, Xiaoying Zhang, Yiling Chen, and Yang Liu. 2024. User-Creator Feature Dynamics in Recommender Systems with Dual Influence. *arXiv preprint arXiv:2407.14094* (2024).

[30] Weiwen Liu, Yunjia Xi, Jiarui Qin, Fei Sun, Bo Chen, Weinan Zhang, Rui Zhang, and Ruiming Tang. 2022. Neural re-ranking in multi-stage recommender systems: A review. *arXiv preprint arXiv:2202.06602* (2022).

[31] María Luisa Menéndez, JA Pardo, L Pardo, and MC Pardo. 1997. The jensen-shannon divergence. *Journal of the Franklin Institute* 334, 2 (1997), 307–318.

[32] Martin Mladenov, Elliot Creager, Omer Ben-Porat, Kevin Swersky, Richard Zemel, and Craig Boutilier. 2020. Optimizing long-term social welfare in recommender systems: A constrained matching approach. In *International Conference on Machine Learning*. PMLR, 6987–6998.

[33] Martin Mladenov, Chih-Wei Hsu, Vihan Jain, Eugene Ie, Christopher Colby, Nicolas Mayoraz, Hubert Pham, Dustin Tran, Ivan Vendrov, and Craig Boutilier. 2021. Recsim ng: Toward principled uncertainty modeling for recommender ecosystems. *arXiv preprint arXiv:2103.08057* (2021).

[34] Marco Morik, Ashudeep Singh, Jessica Hong, and Thorsten Joachims. 2020. Controlling fairness and bias in dynamic learning-to-rank. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 429–438.

[35] Sendhil Mullainathan and Richard H Thaler. 2000. Behavioral economics.

[36] Mohammadmehdi Naghiaei, Hossein A Rahmani, and Yashar Deldjoo. 2022. Cpfair: Personalized consumer and producer fairness re-ranking for recommender systems. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 770–779.

[37] Tien T Nguyen, Pik-Mai Hui, F Maxwell Harper, Loren Terveen, and Joseph A Konstan. 2014. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *Proceedings of the 23rd international conference on World wide web*. 677–686.

[38] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems* 35 (2022), 27730–27744.

[39] Gourab K Patro, Arpita Biswas, Niloy Ganguly, Krishna P Gummadi, and Abhijnan Chakraborty. 2020. Fairrec: Two-sided fairness for personalized recommendations in two-sided platforms. In *Proceedings of the web conference 2020*. 1194–1204.

[40] Siddharth Prasad, Martin Mladenov, and Craig Boutilier. 2023. Content prompting: Modeling content provider dynamics to improve user welfare in recommender ecosystems. *arXiv preprint arXiv:2309.00940* (2023).

[41] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2012. BPR: Bayesian personalized ranking from implicit feedback. *arXiv preprint arXiv:1205.2618* (2012).

[42] Yuta Saito, Himan Abdollahpouri, Jesse Anderton, Ben Carterette, and Mounia Lalmas. 2024. Long-term Off-Policy Evaluation and Learning. In *Proceedings of the ACM on Web Conference 2024*. 3432–3443.

[43] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[44] Reinhard Selten. 1990. Bounded rationality. *Journal of Institutional and Theoretical Economics (JITE)/Zeitschrift für die gesamte Staatswissenschaft* 146, 4 (1990), 649–658.

[45] Chenglei Shen, Xiao Zhang, Wei Wei, and Jun Xu. 2023. HyperBandit: Contextual Bandit with Hypernewtork for Time-Varying User Preferences in Streaming Recommendation. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*. 2239–2248.

[46] Jing-Cheng Shi, Yang Yu, Qing Da, Shi-Yong Chen, and An-Xiang Zeng. 2019. Virtual-taobao: Virtualizing real-world online retail environment for reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 4902–4909.

[47] Sonya Yan Song and Steven S Wildman. 2012. Evolution of strategy and commercial relationships for social media platforms: The case of YouTube. In *Handbook of Social Media Management: Value Chain and Business Models in Changing Media Markets*. Springer, 619–632.

[48] Laura Stein. 2013. Policy and participation on social media: The cases of YouTube, Facebook, and Wikipedia. *Communication, Culture & Critique* 6, 3 (2013), 353–371.

[49] Aixin Sun. 2023. Take a fresh look at recommender systems from an evaluation standpoint. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2629–2638.

[50] Alex Tamkin, Miles Brundage, Jack Clark, and Deep Ganguli. 2021. Understanding the capabilities, limitations, and societal impact of large language models. *arXiv preprint arXiv:2102.02503* (2021).

[51] Choon Hui Teo, Houssam Nassif, Daniel Hill, Sriram Srinivasan, Mitchell Goodman, Vijai Mohan, and SVN Vishwanathan. 2016. Adaptive, personalized diversity

for visual discovery. In *Proceedings of the 10th ACM conference on recommender systems*. 35–38.
[52] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971* (2023).
[53] Lei Wang, Jingsen Zhang, Hao Yang, Zhiyuan Chen, Jiakai Tang, Zeyu Zhang, Xu Chen, Yankai Lin, Ruihua Song, Wayne Xin Zhao, et al. 2023. User behavior simulation with large language model based agents. *arXiv preprint arXiv:2306.02552* (2023).
[54] Yifan Wang, Weizhi Ma, Min Zhang, Yiqun Liu, and Shaoping Ma. 2023. A survey on the fairness of recommender systems. *ACM Transactions on Information Systems* 41, 3 (2023), 1–43.
[55] Stanisław Węglarczyk. 2018. Kernel density estimation and its application. In *ITM web of conferences*, Vol. 23. EDP Sciences, 00037.
[56] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.
[57] Cort J Willmott and Kenji Matsuura. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research* 30, 1 (2005), 79–82.
[58] Yao Wu, Jian Cao, Guandong Xu, and Yudong Tan. 2021. Tfrom: A two-sided fairness-aware recommendation model for both customers and providers. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1013–1022.
[59] Chen Xu, Sirui Chen, Jun Xu, Weiran Shen, Xiao Zhang, Gang Wang, and Zhenhua Dong. 2023. P-MMF: Provider Max-min Fairness Re-ranking in Recommender System. In *Proceedings of the ACM Web Conference 2023*. 3701–3711.
[60] Renzhe Xu, Haotian Wang, Xingxuan Zhang, Bo Li, and Peng Cui. 2024. PPA-Game: Characterizing and Learning Competitive Dynamics Among Online Content Creators. *arXiv preprint arXiv:2403.15524* (2024).
[61] Fan Yao, Chuanhao Li, Denis Nekipelov, Hongning Wang, and Haifeng Xu. 2023. How Bad is Top-K Recommendation under Competing Content Creators?. In *International Conference on Machine Learning*. PMLR, 39674–39701.
[62] Fan Yao, Yiming Liao, Mingzhe Wu, Chuanhao Li, Yan Zhu, James Yang, Jingzhou Liu, Qifan Wang, Haifeng Xu, and Hongning Wang. 2024. User welfare optimization in recommender systems with competing content creators. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 3874–3885.
[63] Fan Yao, Yiming Liao, Mingzhe Wu, Chuanhao Li, Yan Zhu, James Yang, Qifan Wang, Haifeng Xu, and Hongning Wang. 2024. User Welfare Optimization in Recommender Systems with Competing Content Creators. *arXiv preprint arXiv:2404.18319* (2024).
[64] Wenlin Yao, Haitao Mi, and Dong Yu. 2024. HDFlow: Enhancing LLM Complex Problem-Solving with Hybrid Thinking and Dynamic Workflows. *arXiv preprint arXiv:2409.17433* (2024).
[65] Eva Zangerle and Christine Bauer. 2022. Evaluating recommender systems: survey and framework. *Comput. Surveys* 55, 8 (2022), 1–38.
[66] Ruohan Zhan, Konstantina Christakopoulou, Ya Le, Jayden Ooi, Martin Mladenov, Alex Beutel, Craig Boutilier, Ed Chi, and Minmin Chen. 2021. Towards content provider aware recommender systems: A simulation study on the interplay between user and provider utilities. In *Proceedings of the Web Conference 2021*. 3872–3883.
[67] An Zhang, Yuxin Chen, Leheng Sheng, Xiang Wang, and Tat-Seng Chua. 2024. On generative agents in recommendation. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1807–1817.
[68] Fuguo Zhang, Qihua Liu, and An Zeng. 2017. Timeliness in recommender systems. *Expert Systems with Applications* 85 (2017), 270–278.
[69] Guangping Zhang, Dongsheng Li, Hansu Gu, Tun Lu, Li Shang, and Ning Gu. 2023. Simulating news recommendation ecosystem for fun and profit. *arXiv preprint arXiv:2305.14103* (2023).
[70] Xiaoqing Zhang, Xiuying Chen, Yuhan Liu, Jianzhou Wang, Zhenxing Hu, and Rui Yan. 2024. A large-scale time-aware agents simulation for influencer selection in digital advertising campaigns. *arXiv preprint arXiv:2411.01143* (2024).
[71] Guorui Zhou, Xiaoqiang Zhu, Chenru Song, Ying Fan, Han Zhu, Xiao Ma, Yanghui Yan, Junqi Jin, Han Li, and Kun Gai. 2018. Deep interest network for click-through rate prediction. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*. 1059–1068.
[72] Lixin Zou, Long Xia, Zhuoye Ding, Jiaxing Song, Weidong Liu, and Dawei Yin. 2019. Reinforcement learning to optimize long-term user engagement in recommender systems. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*. 2810–2818.