User Welfare Optimization in Recommender Systems with Competing Content Creators

Fan Yao
University of Virginia
Charlottesville, USA
fy4bc@virginia.edu

Yiming Liao
Meta Platforms, Inc.
New York, USA
yimingliao@meta.com

Mingzhe Wu
University of Southern California
Los Angeles, USA
mingzhew@usc.edu

Chuanhao Li
Yale University
New Haven, USA
chuanhao.li.cl2637@yale.edu

Yan Zhu
Google
Mountain View, USA
yanzhuyz@google.com

James Yang
Meta Platforms, Inc.
Menlo Park, USA
jamesjy@meta.com

Jingzhou Liu
Meta Platforms, Inc.
Menlo Park, USA
jingzhol@meta.com

Qifan Wang
Meta Platforms, Inc.
Menlo Park, USA
wqfcr@meta.com

Haifeng Xu
University of Chicago
Chicago, USA
haifengxu@uchicago.edu

Hongning Wang
University of Virginia
Charlottesville, USA
hw5x@virginia.edu

ABSTRACT

Driven by the new economic opportunities created by the creator economy, an increasing number of content creators rely on and compete for revenue generated from online content recommendation platforms. This burgeoning competition reshapes the dynamics of content distribution and profoundly impacts long-term user welfare on the platform. However, the absence of a comprehensive picture of global user preference distribution often traps the competition, especially the creators, in states that yield sub-optimal user welfare. To encourage creators to best serve a broad user population with relevant content, it becomes the platform's responsibility to leverage its information advantage regarding user preference distribution to accurately signal creators.

In this study, we perform system-side user welfare optimization under a competitive game setting among content creators. We propose an algorithmic solution for the platform, which dynamically computes a sequence of weights for each user based on their satisfaction of the recommended content. These weights are then utilized to design mechanisms that adjust the recommendation policy or the post-recommendation rewards, thereby influencing creators' content production strategies. To validate the effectiveness of our proposed method, we report our findings from a series of experiments, including: 1. a proof-of-concept negative example illustrating how creators' strategies converge towards sub-optimal states without platform intervention; 2. offline experiments employing our proposed intervention mechanisms on diverse datasets; and 3. results from a three-week online experiment conducted on Instagram Reels short-video recommendation platform.

CCS CONCEPTS

• **Information systems** → **Personalization**; • **Theory of computation** → **Algorithmic mechanism design**.

KEYWORDS

Recommender System, Mechanism Design, Welfare Optimization

ACM Reference Format:
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 (KDD '24), August 25–29, 2024, Barcelona, Spain.* ACM, Barcelona, Spain, 12 pages. https://doi.org/10.1145/3637528.3672021

1 INTRODUCTION

Online content recommendation platforms have evolved into an indispensable component of our daily lives [6]. These platforms play a pivotal role in assisting their users in navigating the vast ocean of content generated by revenue-seeking creators, including various social media platforms (e.g., Facebook, Instagram), streaming services (e.g., YouTube, TikTok), and many more. One of the primary functions of these recommendation platforms is to advance user welfare, defined as the overall volume and quality of interactions between users and content. This metric is widely regarded as a fundamental indicator of the well-being of an online ecosystem and is also closely tied to the platform's revenue.

KDD '24, August 25–29, 2024, Barcelona, Spain

ACM ISBN 979-8-4007-0490-1/24/08.
https://doi.org/10.1145/3637528.3672021

After decades of effort in relevance-driven matching between users and content, industry practitioners and researchers have reached the consensus that user welfare optimization cannot be achieved through myopic approaches that merely target at eliciting and predicting user preferences [5, 7, 10, 23, 27, 30, 31, 35]. One primary reason is because any matching strategy has a profound impact on content creators' beliefs about the users' demand and consequently their reactions, i.e., what to produce next, leading to a shift in the distribution of content available for recommendation.

This influence pathway is unfortunately overlooked in existing recommendation algorithm design; and therefore, there is a great need for a robust recommendation strategy that operates with respect to creators' strategic responses and the resultant content dynamics. It is imperative for the platform to encourage creators in generating content that continuously contributes to the overall health of the ecosystem.

Typically, creators' well-being is intricately linked to the exposure of their content and the economic incentives they accrue from the platform, compelling them to continuously strive for maximized benefits [13, 15]. This dynamic creates a competitive environment that leads to intriguing phenomena in terms of welfare guarantees at equilibrium [12, 20, 36]. For instance, Yao et al. [32] introduced a game theoretical framework to investigate competition dynamics among content creators. Their research revealed that social welfare loss can be attributed to factors such as the degree of exploration in users' decision making and the span of recommendation slots. As indicated by many previous studies, the platform suffers from sub-optimal social welfare and thus undermines long-term revenue when content distribution lacks necessary diversity to cater to various users' preferences. This issue is also observed in empirical studies, where content creators often exhibit a tendency to chase trends [16, 24]. In essence, creators tend to produce content that arouses the interests of the majority user group, owing to the group's high visibility and the creators' myopic creation strategies [20, 32]. However, it is our contention that the platform should not simply blame creators for their perceived selfishness and myopia. This is because creators do not possess a holistic view of the demand distribution, i.e., user preferences. Instead, it is the platform's responsibility to *disseminate* knowledge about user demand to creators. By doing so, creators can make better informed decisions that mutually benefit their own interest and enhance user welfare (and hence platform's revenue).

In this study, we extend the Content Creator Competition (C^3) framework introduced by Yao et al. [32, 34], to model the dynamics of competition among content creators. We relax the behavioral assumptions about creators' updating strategies in the original framework and explore how the platform can design mechanisms to optimize user welfare accordingly. Our key idea is to direct creators' attention towards currently under-served users, by manipulating creators' received utilities with respect to the cumulative user satisfaction about the recommended content. We present a series of approaches to implement the interventions with theoretical justifications.

To validate the effectiveness of our approach, we conducted offline experiments using both synthetic data and the MovieLens dataset, and demonstrated how our mechanism improves user welfare over time under a creator response simulator. Additionally, we deployed an online experiment on Instagram Reels, a leading short-video recommendation platform, over a span of three weeks and observed statistically significant and positive result in terms of the overall user engagement and content diversity. Our model and online experiments offer valuable insights into the design of incentive-aware recommender platforms. To summarize, our contributions can be listed as follows:

(1) We formalize the user welfare optimization problem in a competitive content creation environment and identify the primary cause for potential sub-optimal outcomes: the information asymmetry between content creators and the platform.
(2) We propose a dynamic user importance reweighting approach with theoretical justifications for optimizing user welfare and three implementation schemes which can be applied to various practical scenarios.
(3) We demonstrate the effectiveness of our solution with both offline simulations and online testing on real traffic.

2 RELATED WORK

The characterization and optimization of long-term dynamics on content platforms involving strategic content creators has garnered increasing attention from both theoretical [3, 4, 9, 17–19, 19, 20, 29, 32–34, 36] and empirical [23, 26] fields. Seminal works from Ben-Porat and Tennenholtz [3, 4] introduced a game theoretical setting to model interactions between content creators and users, and proposed the Shapley mediator to ensure the existence of a pure Nash Equilibrium [25].

Recently, Yao et al. [32] demonstrated that due to creators' competition, the user welfare loss under a top-K recommender systems can be upper-bounded by $O(\frac{1}{\log K})$. This finding suggests that the platform can improve user welfare by providing more recommendations. Building on this, the authors further proposed a category of mechanisms for the platform to ensure a stable equilibrium and developed a computational solution to identify the optimal mechanism for social welfare optimization [34]. Additionally, Zhu et al. [36] introduced an online learning method to jointly optimize recommendation policy and payment contracts for creators to maximize accumulated utility. Hu et al. [18] designed a learning algorithm to incentivize the creation of high-quality content. However, all these studies rely on strong behavioral assumptions about content creators, e.g., they can perform no-regret learning [32], or have oracle access to their utility functions [3, 4, 34], so that the Nash equilibium is achievable. Our work bridges this gap by developing a system-side solution to optimize user welfare that even when creators are not able to achieve Nash equilibria.

On the empirical side, Mladenov et al. [23] explored a scenario where content creators may leave the platform if their user engagement falls below a threshold. The study optimized social welfare by solving a constrained matching problem. In a similar spirit, Prasad et al. [26] introduced a sequential prompting policy aimed at optimizing user welfare in equilibrium. The optimal policy was determined through mixed integer programming. The solutions were reported to be effective under specific behavioral assumptions or environmental contexts, e.g., the platform can send prompts to creators as additional signals. However, the platforms are often constrained in their ability to influence the ecosystem. They may

primarily rely on monetary incentives to motivate creators and have limited flexibility to manipulate factors beyond matching strategies and post-matching rewards. Our solution addresses this broader range of scenarios, making it applicable, for example, when creators are highly responsive to monetary incentives, and the platform's influence is primarily exerted through adjustments to matching probabilities and post-matching rewards.

3 THE MODELING OF CONTENT CREATION COMPETITION

In this section, we formulate the competition among content creators (i.e., players) as a strategic game, which will serve as an environment for the subsequent mechanism design problem. At a high level, each creator's utility is determined by the platform's matching strategy and the post-matching reward function. Creators adhere to simple, local update principles to sequentially alter their strategies, resulting in a dynamic content distribution on the platform. The primary objective of the platform is to optimize the cumulative user welfare by designing its matching strategy and post-matching reward function. Our strategic game setup builds upon and extends the framework of Content Creator Competition (C^3) game introduced in [32, 34]. For the sake of simplicity in nomenclature, we retain the name of C^3 and refer to our game as C^3_{ext}, i.e., an extension of the C^3. Formally, a C^3_{ext} instance is defined by the following tuple: $\left(\mathcal{X}, \{\mathcal{S}_i\}_{i=1}^n, \sigma, \beta, K, R(\cdot)\right)$, which we explain in details below.

(1) **Basic setups:** a user distribution $\mathcal{X}$ with finite support $\{\boldsymbol{x}_j \in \mathbb{R}^d\}_{j=1}^m$, and a set of content creators denoted by $[n] = \{1, \cdots, n\}$. Each creator i can take an action $\boldsymbol{s}_i$, is often referred to as a *pure strategy* in game-theoretic literature, from an action set $\mathcal{S}_i \subset \mathbb{R}^d$. $\boldsymbol{s}_i$ can be understood as the embedding of content that creator i will produce. Without loss of generality, we assume the L_2 norms of any $\boldsymbol{x}$ and $\boldsymbol{s}_i$ are upper bounded by 1.

(2) **Relevance function:** the relevance function $\sigma(\boldsymbol{s}, \boldsymbol{x}) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_{\geq 0}$ measures the *relevance* score between a user $\boldsymbol{x} \sim \mathcal{X}$ and content $\boldsymbol{s}$. Without loss of generality, we normalize σ to $[0, 1]$, where 1 suggests perfect matching. We focus on modeling the strategic behavior of creators and thus abstract away the estimation of σ [1]. For simplicity, we use $\sigma_{i,\boldsymbol{x}}$ to denote $\sigma(\boldsymbol{s}_i, \boldsymbol{x})$ when the joint strategy profile $\boldsymbol{s} = (\boldsymbol{s}_1, \cdots, \boldsymbol{s}_n) \in \mathcal{S}$ and user profile $\boldsymbol{x}$ are clear in the context of our discussion.

(3) **Matching function:** Given any user $\boldsymbol{x} \in \mathcal{X}$ and when each creator commits to a strategy $\boldsymbol{s}_i$, the platform retrieves the top-K ranked content in terms of the relevance scores $\{\sigma_{i,\boldsymbol{x}}\}_{i=1}^n$ and match one of them to $\boldsymbol{x}$. Specifically, let $\{\sigma_{l(1),\boldsymbol{x}} \geq \cdots \geq \sigma_{l(n),\boldsymbol{x}}\}$ be a permutation of $\{\sigma_{i,\boldsymbol{x}}\}_{i=1}^n$, we assume that the platform would pick $\boldsymbol{s}_{\boldsymbol{x}} \in L_{\boldsymbol{x}}(K; \boldsymbol{s}) \triangleq \{\sigma_{l(i),\boldsymbol{x}}\}_{i=1}^K$ using a softmax distribution with temperature $\beta \geq 0$ [2], i.e.,

$$P_i(\boldsymbol{s}, \boldsymbol{x}) \triangleq Prob[\boldsymbol{s}_{\boldsymbol{x}} = \boldsymbol{s}_{l(i)}] \propto \exp[\beta^{-1}\sigma_{l(i),\boldsymbol{x}}], 1 \leq i \leq K. \quad (1)$$

A small β makes the matching strategy more deterministic, and $\beta \to \infty$ corresponds to random matching.

(4) **User utility and welfare:** When user $\boldsymbol{x}$ is matched with $\boldsymbol{s}$, the user's perceived utility is given by a function $\pi(\boldsymbol{s}, \boldsymbol{x})$. The user welfare $W(\boldsymbol{s})$ is thus defined as the total expected utility resulted from the matching,

$$W(\boldsymbol{s}) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}[\pi(\boldsymbol{s}_{\boldsymbol{x}}, \boldsymbol{x})]. \quad (2)$$

To simplify the technical discussions, we assume the learned relevance function σ is an unbiased estimation of π, and therefore $W(\boldsymbol{s})$ can be simplified to

$$W(\boldsymbol{s}) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}[\sigma(\boldsymbol{s}_{\boldsymbol{x}}, \boldsymbol{x})]. \quad (3)$$

However, our proposed solution works for general welfare function defined in Eq (2).

(5) **Creator utility:** For creator i, her utility is given by

$$u_i(\boldsymbol{s}) = \mathbb{E}_{\boldsymbol{x}\in\mathcal{X}}\left[R(\boldsymbol{s}_i, \boldsymbol{x}) \cdot P_i(\boldsymbol{s}, \boldsymbol{x})\right], \quad (4)$$

where $R(\boldsymbol{s}_i, \boldsymbol{x})$ is the system-provided reward for this matching. Natural choices of R include $R(\boldsymbol{s}_i, \boldsymbol{x})$ being proportional to the user's perceived utility, or simply setting $R(\boldsymbol{s}_i, \boldsymbol{x}) = 1$ (i.e., reward creators by the amount of traffic). Therefore, we have

$$u_i(\boldsymbol{s}) = \mathbb{E}_{\boldsymbol{x}\in\mathcal{X}}\left[\sigma(\boldsymbol{s}_i, \boldsymbol{x}) \cdot P_i(\boldsymbol{s}, \boldsymbol{x})\right], \quad (5)$$

$$u_i(\boldsymbol{s}) = \mathbb{E}_{\boldsymbol{x}\in\mathcal{X}}\left[P_i(\boldsymbol{s}, \boldsymbol{x})\right], \quad (6)$$

Throughout the paper we adopt Eq (5) as the platform's default choice, as it is demonstrated in [32] that rewarding creators by user utility enjoys a better welfare guarantee than rewarding them by traffic.

The most well established concept for characterizing a game's outcome is pure Nash equilibrium (PNE) [25]. At a PNE, any possible deviation from a player's current strategy would not increase her utility conditioned on other players' strategies. Under some mild assumptions, we can prove that the PNE of our C^3_{ext} game exists and is unique as stated in the following theorem.

Theorem 3.1. *Any C^3_{ext} game with $K = n$ has a unique pure Nash equilibrium (PNE) under the utility function (6) if $\sigma(\cdot)$ is sufficiently smooth and concave and each creator has a convex strategy set.*

Theorem 3.1 guarantees the existence of a unique PNE and thus theoretically allows the platform to establish a stable outcome. However, in practical scenarios, we find it uninteresting to either generalize this result or delve further into its properties for two reasons. First, it is rare for $K = n$ to hold in practice because no system will present the entire collection of content to each user. When $K < n$, the existence of a PNE becomes challenging to establish, due to the discontinuity of the utility functions caused by the top-K ranking operator during the matching process. Second, even when a PNE does exist, it does not suggest that creators can consistently reach it through sequential updates. Furthermore, the existence of a PNE does not necessarily imply it is easily achievable in practice, nor does it suggest an improved user welfare. In fact, as we will demonstrate in Section 4.1, even in a simple environment with a unique PNE, a natural updating dynamics among creators fails to converge to the PNE and results in sub-optimal user welfare.

Therefore, we focus on a more practical solution concept called *Local Nash equilibria (LNE)*. While a PNE requires that all players

[1] We assume σ is learned from the offline data and $\sigma(\boldsymbol{s}_i, \boldsymbol{x})$ is an unbiased estimation of user $\boldsymbol{x}$'s satisfaction when exposed to $\boldsymbol{s}_i$.

[2] The formulation in [32] also assumes the platform retrieve top-K content for each user, but let the user to choose one according to the Random Utility model. The resulting matching probability shares the same form as in Eq. (1), but differs in the sense that the β in our setting is a parameter controlled by the platform while it is the user decision noise in [32].

do not want to deviate to any other strategy in the entire space, an LNE merely stipulates players are satisfied with their strategies in a local region. Its formal definition is given as follows.

Definition 3.2. A profile of creator strategies $\{s_i^*\}_{i=1}^n$ *forms a local Nash equilibrium (LNE), if for every creator* i*, there exists an open set* $\mathcal{S}_i^0 \in \mathcal{S}_i$ *such that* s_i^* *is a best response strategy within* $\mathcal{S}_i^0$*; formally,*

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*) \quad \text{for every } s_i \in \mathcal{S}_i^0. \tag{7}$$

We argue that LNE offers a more intuitive and practical solution concept for consideration due to two observations. First, the strategic evolution of content creation is often deeply intertwined with creators' historical decisions [21]. This correlation stems from content generation being anchored in domain-specific expertise and accumulated experiences, which are inherently stable attributes. As a result, the produced content usually demonstrates path dependency, posing significant challenges for creators in implementing drastic modifications. Second, creators are typically constrained by a lack of comprehensive insights into their utility functions due to a limited understanding of the user demographic and the distribution of user preferences. Given these constraints, creators are likely to resort to incremental adjustments for strategy update.

Hence, we focus on the setting where creators engage in a repeated play of C_{ext}^3 and employ a local searching rule termed local better response (LBR) update for improving their strategies. The details of LBR is presented in Algorithm 2 in Appendix A. LBR characterizes two fundamental properties of content creation: 1. it relies solely on point estimations of the utility function; and 2. it only incurs local changes at each update. At each step, a creator who decides to update her strategy would first generate an exploration direction $\boldsymbol{g}_i$ and then she would evaluate whether adjusting her strategy in this direction results in a higher utility. If so, she proceeds to update her strategy along $\boldsymbol{g}_i$ in a pace of η; otherwise, she maintains her current strategy. This procedure closely emulates real-world scenarios where creators strive to optimize their utilities while having merely black-box access to the utility functions. In practice, finding a clear direction that guarantees improved utility can be a challenging and, at times, unrealistic task. Consequently, we model their strategy evolution as an iterative process of trial and error. By definition, when LBR converges in C_{ext}^3, it must converge to an LNE. Our primary interest lies in understanding how the platform can devise a dynamic rewarding or matching principle that maximizes cumulative user welfare within a given time period.

4 INTERVENTION MECHANISM DESIGN

In this section, we introduce the new intervention mechanism designed to optimize user welfare. These mechanisms are intended for the platform to influence creators' perceived utilities, thereby guiding the evolution of their strategies toward more desirable outcomes. We will first establish the need for platform-driven mechanism design by illustrating how suboptimal results can arise in a simplified example without any intervention. Subsequently, we will delve into the specifics of our proposed methods.

4.1 The Necessity of Intervention

We start with a simple illustrative example to show how the competition among creators could result in quite inferior user welfare

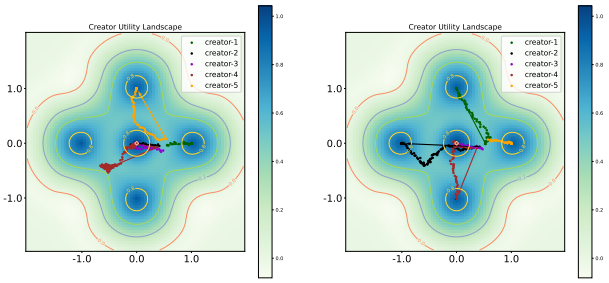


Figure 1: Visualization of creators' evolving strategies. Left: no intervention, right: platform decreases the weight of the center user by half. Creators' strategies are marked with different colors, and the arrows start from initial strategies and point to the last-iterate strategies.

in C_{ext}^3 when creators employ local update dynamics specified in Algorithm 2. This example exhibits a stark contrast to the sound welfare guarantee for no-regret learning [2] equipped creators in [32]. Consider a C_{ext}^3 instance $(\mathcal{X}, \{\mathcal{S}_i\}_{i=1}^n, \sigma, \beta, K, R(\cdot))$ described below. The user population $\mathcal{X}$ is evenly distributed over the finite set $\{\boldsymbol{x}_j\}_{j=1}^5 = \{(0,0),(1,0),(0,1),(-1,0),(0,-1)\}$ and there are $n = 5$ content creators, each with action set $\mathcal{S}_i = \mathbb{R}^2$. The reward function is defined as $R(\boldsymbol{s}_i, \boldsymbol{x}) = \sigma(\boldsymbol{s}_i, \boldsymbol{x}) = \max\{2 - \|\boldsymbol{s}_i - \boldsymbol{x}\|_2, 0\}$ and $\beta = 10, K = 3$. It is evident that the user welfare defined in Eq (3) is maximized when each creator precisely targets a single user, i.e., $\boldsymbol{s}_i = \boldsymbol{x}_i, 1 \leq i \leq 5$, which also represents the PNE of this game. However, as we will illustrate through simulations, creators' strategies do not converge to the PNE nor optimize the user welfare under the LBR dynamics when the platform does not intervene.

First, let's examine what happens when the platform takes no action to guide the creators. The left panel of Figure 1 visualizes the trajectories of strategy evolution in our constructed environment. Initially, creators' strategies are randomly distributed in the region between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. Over time, $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$ are exclusively occupied by one creator each, while $\boldsymbol{x}_1$ has two creators competing for it. The remaining creator chooses not to target either $\boldsymbol{x}_4$ or $\boldsymbol{x}_5$ and hovers around the region between $\boldsymbol{x}_4$ and $\boldsymbol{x}_5$, leaving both $\boldsymbol{x}_4$ and $\boldsymbol{x}_5$ unsatisfied.

From the observed strategy evolution paths, we can deduce how this sub-optimal situation arises. Initially, creators move in different directions: two creators quickly converge to $\boldsymbol{x}_2$ and $\boldsymbol{x}_3$, while the remaining three compete for the attention of the central user $\boldsymbol{x}_1$. However, after this point, no creator has a strong incentive to move closer to $\boldsymbol{x}_4$ or $\boldsymbol{x}_5$, as the marginal utility gained from getting closer to $\boldsymbol{x}_4$ or $\boldsymbol{x}_5$ does not compensate for the loss incurred by moving away from $\boldsymbol{x}_1$. Consequently, two creators decide to remain around $\boldsymbol{x}_1$ and one creator settles in a region between $\boldsymbol{x}_4$ and $\boldsymbol{x}_5$.

The above observations highlight the pivotal role played by the central user $\boldsymbol{x}_1$ in the occurrence of sub-optimal results. Since $\boldsymbol{x}_1$ is close to other users in the embedding space, targeting $\boldsymbol{x}_1$ becomes a popular and safe choice for creators. It secures a fraction of attention from $\boldsymbol{x}_1$ without completely sacrificing the utility gained from other user groups. Thus, users like $\boldsymbol{x}_1$ act as "popular states" when creators dynamically adjust their strategies. Whenever a creator is located near $\boldsymbol{x}_1$, they are likely to be trapped and reluctant to explore

potentially better strategies. Consequently, such "popular" users end up attracting more creators, leaving other users unattended.

One immediate solution for the platform is to identify and reduce the impact of these "popular" users. For instance, the platform can halve the utility gained from the central user $\boldsymbol{x}_1$ for each creator. This simple mechanism works effectively in this example, as illustrated in the right panel of Figure 1. Initially, there are still three creators converging to $\boldsymbol{x}_1$. However, due to the reduced reward from $\boldsymbol{x}_1$, two creators find it less profitable to stay, driving them to deviate towards $\boldsymbol{x}_4$ and $\boldsymbol{x}_5$. By assigning different importance weight for each user, the platform can reshape each creator utility landscape and therefore influence their local search based dynamical behaviors.

4.2 Platform's Intervention Mechanisms

The observations above motivate our design of intervention mechanisms that can be employed by the platform to influence creators' perceived utilities. These mechanisms lay the foundation for the adaptive optimization methods we will delve into later. As a reminder, as defined in Eq (4), a creator's expected utility from a specific user $\boldsymbol{x}$ is influenced by two key factors: the probability of creator i being matched with user $\boldsymbol{x}$ denoted as $P_i(\boldsymbol{s}, \boldsymbol{x})$, and the post-matching reward assigned by the platform, denoted as $R(\boldsymbol{s}_i, \boldsymbol{x})$. The default choice of the platform is to set the reward function $R(\boldsymbol{s}_i, \boldsymbol{x}) = \sigma(\boldsymbol{s}_i, \boldsymbol{x})$ as in Eq (5) and the matching probability function $P_i(\boldsymbol{s}, \boldsymbol{x})$ as the softmax over the top-K ranked content $\boldsymbol{s}_i$ as demonstrated in Eq (1).

In the example provided in Section 4.1, the primary factors leading to sub-optimal welfare is the presence of popular user groups that attract excessive creator attention, making minority user groups unnoticed by creators. To enhance overall user welfare, it is crucial for the platform to guide creators' attention toward these overlooked user groups by re-emphasizing their significance. In this way, creators who were previously unaware of these user groups or found them less lucrative may consider adjusting their strategies to align more closely with those users' preferences. To achieve this objective, we introduce and study three different approaches for modifying the schemes of $R(\boldsymbol{s}_i, \boldsymbol{x})$ and $P_i(\boldsymbol{s}, \boldsymbol{x})$, namely User Importance Reweighting (UIR), Soft Matching Truncation (SMT), and Hard Matching Truncation (HMT). These three mechanisms share a common underlying principle, but they are designed to operate under different scenarios, taking into account potential constraints faced by a platform.

User Importance Reweighting (UIR) The most straightforward approach is UIR,

$$u_i(\boldsymbol{s}_i, \boldsymbol{s}_{-i}) = \mathbb{E}_{\boldsymbol{x}\in\mathcal{X}}[w(\boldsymbol{x}) \cdot R(\boldsymbol{s}_i, \boldsymbol{x}) \cdot P_i(\boldsymbol{s}, \boldsymbol{x})], \tag{8}$$

where the platform simply adjusts the post-matching rewards for creators based on the measured importance of each user. Specifically, if the platform believes a user has been under-served under the current content distribution, it raises the reward for creators whose content is consumed by such a user. Intuitively, this sends a message to creators that "if you shift your content towards such users, you will get a higher marginal reward compared to sticking to your current content." As a result, the platform can carefully design the user weights such that a reasonable number of creators can be successfully incentivized to serve the targeted users.

Soft Matching Truncation (SMT) and Hard Matching Truncation (HMT) Both SMT and HMT function in a similar manner as UIR but focus on manipulating the matching probability rather than the post-matching reward by utilizing the weight $w(\boldsymbol{x})$. Recall that the probabilistic matching function P is characterized by two parameters: the truncation number K (which, in practice, corresponds to the total number of recommendation candidates retrieved for ranking) and the temperature β (which can be viewed as a measure of the exploration strength in the ranking model). When the platform needs to signal the importance of a specific user $\boldsymbol{x}$, it enhances $\boldsymbol{x}$'s visibility among creators, increasing the chance that creators who were previously unaware of $\boldsymbol{x}$ start realizing the potential benefits of catering to $\boldsymbol{x}$. This can be achieved by either increasing β or K: increasing β flattens the distribution of $\boldsymbol{x}$'s matches among the top-K candidates, while increasing K enlarges the pool of creators exposed to $\boldsymbol{x}$. Therefore, both of them augment the expected number of creators exposed to $\boldsymbol{x}$. Since K imposes a rigid threshold on the number of creators exposed to $\boldsymbol{x}$, while β offers a more flexible threshold, we refer to them as Hard Matching Truncation (HMT) and Soft Matching Truncation (SMT), respectively:

$$u_i(\boldsymbol{s}_i, \boldsymbol{s}_{-i}) = \mathbb{E}_{\boldsymbol{x}\in\mathcal{X}}[R(\boldsymbol{s}_i, \boldsymbol{x}) \cdot P_i(\boldsymbol{s}, \boldsymbol{x}; \beta(w(\boldsymbol{x})), K)], \tag{9}$$

$$u_i(\boldsymbol{s}_i, \boldsymbol{s}_{-i}) = \mathbb{E}_{\boldsymbol{x}\in\mathcal{X}}[R(\boldsymbol{s}_i, \boldsymbol{x}) \cdot P_i(\boldsymbol{s}, \boldsymbol{x}; \beta, K(w(\boldsymbol{x})))]. \tag{10}$$

We remark that UIR is more suitable when the platform possesses the flexibility to design payment incentives for creators. However, if the platform has limited control over payment, such as budget constraints or other factors, SMT or HMT can be employed, as they only require minor adjustments to the matching function. The specific choices of increasing functions $\beta(\cdot), K(\cdot)$ are flexible and we leave it to the experiments.

4.3 Welfare Optimization through Adaptive Reweighing

To implement our proposed intervention mechanisms, we need to compute the corresponding user-specific weighting functions, namely $w(\cdot)$, $\beta(\cdot)$, and $K(\cdot)$. In this section we will use UIR as an example to illustrate our method and let the user distribution $\mathcal{X}$ be a uniform distribution over its support $\{\boldsymbol{x}_1, \cdots, \boldsymbol{x}_m\}$ so that $w(\cdot)$ can be parameterized by a vector $\boldsymbol{w} \in \mathbb{R}^m_{\geq 0}$. When the platform commits to an intervention mechanism $\boldsymbol{w}$, the content creators' strategic updates according to LBR (i.e., algorithm 2) will lead their joint strategy to an LNE $\boldsymbol{s}^*$, which determines the content distribution and the total user welfare W. Therefore, the task of finding the optimal $\boldsymbol{w}$ maximizing W under C^3_{ext} can be formulated as the following bi-level optimization problem:

$$\max_{\boldsymbol{w}\in\mathbb{R}^m_{\geq 0}} \quad W(\boldsymbol{s}^*(\boldsymbol{w})) \tag{11}$$

$$\text{s.t.,} \quad \boldsymbol{s}^*(\boldsymbol{w}) \text{ is an LNE of } C^3_{\text{ext}}. \tag{12}$$

We adopt the formulation in Eq (11) simply for presentation purpose, as the constraint in Eq (12) is not well-defined due to the non-uniqueness of LNE of C^3_{ext} in general. When we takcle problem in Eq (11), we employ either LBR for simulating an $\boldsymbol{s}^*(\boldsymbol{w})$ in offline experiments, or we directly observe $\boldsymbol{s}^*(\boldsymbol{w})$ based on the creators' actual responses over a period of time for online experiments. An straightforward approach to solve Eq (11) is to use an iterative

method to dynamically adjust $\boldsymbol{w}$, and the main challenge is to pin down an improving direction of $\boldsymbol{w}$. Ideally, we can apply first-order optimization if an estimation of the gradient $\frac{dW}{d\boldsymbol{w}}$ is available. However, the interplay between $\boldsymbol{w}$ and $\boldsymbol{s}^*(\boldsymbol{w})$ is generally intractable to analyze and we have to resort to heuristic methods. To get an intuitive idea about an improving direction of $\boldsymbol{w}$, we consider a stylized setting where the user population is perfectly separated and the relevance function is given by dot-product $\sigma(\boldsymbol{s}, \boldsymbol{x}) = \boldsymbol{s}^\top \boldsymbol{x}$. In such a structured environment, the following theorem reveals a useful principle for finding an improving direction of $\boldsymbol{w}$.

Theorem 4.1. *When the number of creators n is large enough and the user population $\mathcal{X}$ is a uniform distribution over an orthogonal basis in $\mathbb{R}^d$, updating $\boldsymbol{w}$ with the following formula guarantees an improvement in W defined in Eq* (3)*:*

$$w'_j = w_j \cdot e^{-\eta \bar{\pi}(\boldsymbol{x}_j)}, \forall j \in [m], \tag{13}$$

where η is a small scalar denoting the learning rate, and $\bar{\pi}(\boldsymbol{x}_j)$ is the expected utility of user $\boldsymbol{x}_j$ at $\boldsymbol{s}^*(\boldsymbol{w})$.

By the definition in Eq (4), rescaling each w_j by a constant does not alter the nature of problem in Eq (11). Therefore, the insight conveyed by Eq (13) is clear: if a user enjoys a high expected utility under the current content distribution, the platform should reduce her weight when rewarding creators. Conversely, if a user's expected utility is relatively low, the platform needs to highlight her significance for motivating a larger set of creators to develop content that caters to the needs of this user. Despite the fact that Eq (13) is derived from a significantly simplified user distribution, we will leverage it as a foundational element in the development of our adaptive reweighing algorithm and demonstrate in our experiments that this simple heuristic works pretty well for real user distributions.

Next, we formally introduce our proposed adaptive reweighting algorithm for optimizing the intervention mechanism $\boldsymbol{w}$. Each user $\boldsymbol{x}$ is initially assigned a unit weight $\boldsymbol{w}^{(0)}(\boldsymbol{x}) = 1$. During subsequent iterations, the platform continuously monitors the average utility of user $\boldsymbol{x}$, denoted as $\bar{\pi}(\boldsymbol{x})$, within a specified time window, and updates $\boldsymbol{w}$ according to the following (14), where $\alpha > 0$ is a tunable parameter. This adjustment process employs the meta-algorithm structure of multiplicative weight update method [1].

$$w^{(i+1)}(\boldsymbol{x}) \propto w^{(i)}(\boldsymbol{x}) \cdot \exp(-\alpha \bar{\pi}(\boldsymbol{x})). \tag{14}$$

In practice, we can choose the user utility function $\pi(\boldsymbol{x})$ as the metric used for defining the user welfare function Eq (2). Up to this point, our discussion has primarily focused on the assumption that $\pi(\boldsymbol{x}; \boldsymbol{s}) \propto \sigma(\boldsymbol{s}_i, \boldsymbol{x})$. However, it is important to highlight that π in Eq (14) can also take alternative forms to optimize empirical performance. For instance, it can be a function of any numerical measurement related to user satisfaction (e.g., click-through rate). To reduce the dimension of the user weight vector and enhance the robustness of weight updates, we recommend that algorithm designers pre-cluster users into L groups based on their static features so that users within the same group maintain identical weights. The platform's intervention strategy is thus parameterized by an L-dimensional vector, $\boldsymbol{w} = (w_1, \cdots, w_L)$, with each entry denoting the weight assigned to the corresponding user group.

For a fixed time horizon T in which the platform plans to perform intervention, the platform divides the horizon into E epochs, each with an equal length of M (i.e., $T = EM$). At the start of each epoch e, the platform commits to a weight vector $\boldsymbol{w}^{(e)}$ and deploy it to one of the intervention mechanisms UIR, SMT or HMT. After that, the platform observes and records the sequence of creators' strategic responses, denoted as $\{\boldsymbol{s}^{(e,i)}\}_{i=1}^M$ from the online environment. Subsequently, the algorithm estimates the average user welfare $\bar{\pi}_l$ for each group l. It then employs values in $\{\bar{\pi}_l\}_{l=1}^L$ to update the weights at the beginning of the $(e+1)$-th epoch using Eq (14). To prevent $\boldsymbol{w}$ from growing or declining excessively, after each update we first normalize and then clip its values within a predetermined interval $[w_{\min}, w_{\max}]$. The formal description of this process is presented in Algorithm 1. The implementation details about the deployment of UIR, SMT and HMT in Line 4 are deferred to Appendix B.

Algorithm 1 Adaptive Reweighting

1: **Input:** Number of epochs E, Epoch length M, Initial strategy profile $\boldsymbol{s}^{(0)}$, learning rate η, temperature parameter α, user groups $(G_1, \cdots, G_L)$, clipping constant $w_{\min}, w_{\max}$.
2: **Initialization:** Initial weight $\boldsymbol{w}^{(0)} = (w_1^{(0)}, \cdots, w_L^{(0)})$.
3: **for** $e = 0$ **to** E **do**
4: Deploy the weight $\boldsymbol{w}^e$ using UIR (Eq (8)), SMT (Eq (9)) or HMT (Eq (10)).
5: Observe creators' strategy sequence $\{\boldsymbol{s}^{(e,i)}\}_{i=1}^M$.
6: Compute the average user utility for each group

$$\bar{\pi}_l = \frac{1}{M|G_l|} \sum_{\boldsymbol{x} \in G_l} \sum_{i=1}^{M} \pi(\boldsymbol{x}; \boldsymbol{s}^{(e,i)}).$$

7: Update $w_l^{(e+\frac{1}{3})} = w_l^{(e)} \cdot \exp(-\alpha \bar{\pi}_l), l \in [L]$.
8: Normalize $w_l^{(e+\frac{2}{3})} = L \cdot w_l^{(e+\frac{1}{3})} / \sum_{j=1}^L w_j^{(e+\frac{1}{3})}, l \in [L]$.
9: Clip $\boldsymbol{w}^{(e+1)} = \mathbf{Clip}(\boldsymbol{w}^{(e+\frac{2}{3})}, w_{\min}, w_{\max})$.
10: Set $\boldsymbol{s}^{(e+1)} = \boldsymbol{s}^{(e,M)}$.

5 EXPERIMENTS

In this section, we evaluate our proposed intervention mechanisms on both offline datasets and an online environment on a leading short-video recommendation platform in the industry.

5.1 Experiments on Offline Data

We conduct simulations on C^3_{ext} game instances constructed from synthetic data and MovieLens-1m dataset [14]. In the following, we first introduce the specification of these two simulation environments and then report the results.

5.1.1 Synthetic environment. For the synthetic environment, we first construct the user population as follows: we fix an embedding dimension $d = 5$ and independently sample 10 cluster centers, denoted as $\{\mathbf{c}_1, \cdots, \mathbf{c}_{10}\}$, from the unit sphere $\mathbb{S}^{d-1}$. For each center $\mathbf{c}_i$, we generate users belonging to cluster-i by independently sampling from a Gaussian distribution $\mathcal{N}(\mathbf{c}_i, 0.5^2 I_d)$. The sizes of the 10 user clusters are denoted by a vector $\boldsymbol{z} = 10 \times$

$(100, 50, 20, 10, 10, 5, 2, 1, 1, 1)$. In this manner, we generate a population $\mathcal{X}$ of size $m = 2000$. The number of creators is set to $n = 200$, and each action set $\mathcal{S}_i$ is set to the unit ball in $\mathbb{R}^d$. The user utility and relevance score function are set to $\pi(\boldsymbol{s}, \boldsymbol{x}) = \sigma(\boldsymbol{s}, \boldsymbol{x}) = \max\{1 - \|\boldsymbol{s} - \boldsymbol{x}\|/3, 0\}$. We set (β, K) to $(0.1, 20)$ by default. Such synthetic datasets characterize a class of clustered user preference distributions (e.g., majority vs., minority user groups).

On the creators' side, we let their initial strategies to be close the center of the largest user group. This environment models a situation where creators tend to chase popular trends by exclusively producing content tailored to the taste of the largest user group. We aim to investigate whether our proposed mechanisms can assist the platform to escape from such sub-optimal states.

5.1.2 Environment constructed from MovieLens-1m. We use deep matrix factorization [11] to train user and movie embeddings (with dimension set to 32) by fitting the observed ratings in the range of 1 to 5. To ensure the quality of the trained embeddings, we performed a 5-fold cross-validation and obtained an averaged RMSE = 0.739 on the test sets. Then with the same hyper-parameter settings, we train the user/item embeddings with the complete dataset.

We select active users with more than 200 ratings, resulting in a population $\mathcal{X}$ comprising 1578 users. We set the number of creators to 20, with each creator's action set $\mathcal{S}_i$ consisting of 1000 different movies. All $\{\mathcal{S}_i\}$ share a common part – the most popular 700 movies based on the number of ratings they received, and each $\mathcal{S}_i$ also has a private part – a randomly sampled 300 movies. Our choice of the user utility and matching score functions is $\pi(\boldsymbol{s}, \boldsymbol{x}) = \sigma(\boldsymbol{s}, \boldsymbol{x}) = \boldsymbol{s}^\top \boldsymbol{x}$, and then normalized to the region $[0, 1]$. Additionally, we set $(\beta, K) = (0.1, 20)$ and initialize creators' strategies to the most preferred movie among all users (i.e., the movie that enjoys the highest average rating among $\mathcal{X}$).

5.1.3 Configurations of adaptive reweighting algorithm and intervention mechanisms. For the adaptive reweighting algorithm, we set the epoch length $M = 5$ and the simulation time horizon $T = 3000$ for both environments. During each time step within an epoch, we simulate creators' responses by letting each of them update her strategy once using Algorithm 2 in a random order. Creators' learning rate is set to $\eta = 0.2$. On the platform side, we use K-means clustering to determine user groups and set the number of clusters to 20 for synthetic environment and 15 for MovieLens environment, respectively. We should note as in practice, even the system does not have the exact knowledge about user distribution, we do not use the ground-truth clustering of users set in the simulation. In addition, we set the temperature parameter $\alpha = 0.5$ for the first half of the time period and reduce it to 0.1 for the remaining period. The clipping constants are set to $(w_{\min}, w_{\max}) = (0.2, 5.0)$ and the mapping used in SMT and HMT are set to $\beta(\boldsymbol{x}) = \beta \cdot w(\boldsymbol{x})$ and $K(\boldsymbol{x}) = \lceil K \cdot w(\boldsymbol{x}) \rceil$.

5.1.4 Results. Figure 2a illustrates the user welfare resulted from creators' evolving strategies under the three intervention mechanisms: UIR, SMT, and HMT, compared to the baseline (no platform intervention). Over time, all three mechanisms consistently outperform the baseline. In the baseline (shown in blue), the welfare plateaus quickly and remains stagnant. Conversely, the welfare curves under the other mechanisms exhibit "double-ascent" patterns. Initially, they also plateau, but eventually, they begin to rise again and surpass the baseline. This is because, without platform's intervention, creators tend to remain in sub-optimal equilibria as illustrated in Section 4.1. However, our proposed mechanisms gradually accumulate user group weights, which, when significant enough, encourage creators to explore unattended user groups, leading to increased welfare. Among the three mechanisms, HMT demonstrates the most substantial gain with the least variance. UIR, while showing a lower marginal gain, maintains stability with minimal variance. SMT, which achieves a moderate gain, exhibits higher variance, suggesting that directly manipulating the matching temperature may be overly aggressive.

Figure 2b shows the learned group weights at the last iteration of simulation. As it demonstrates, all three mechanisms emphasize on small groups over larger ones. This outcome aligns with our expectation: on one hand, larger user groups are more likely to "trap" unnecessarily many creators and thus should be deprioritized; on the other hand, increasing weights of niche user groups also improve their chances of being discovered by more creators.

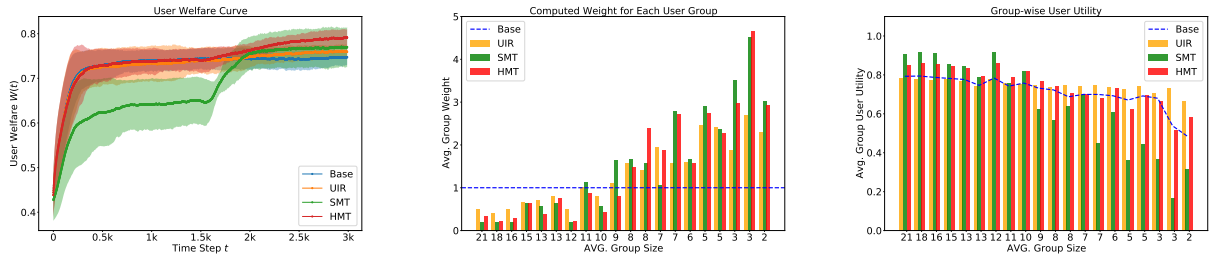
Figure 2c breaks down the average utilities across user groups. The blue dashed line (i.e., the no-intervention baseline) exhibits a positive correlation between averaged group utility and group size, mirroring real-world observations. The orange bars show that UIR strikes a balance by improving the utility of niche groups while slightly trading off utility in larger groups. HMT achieves a remarkable Pareto improvement across all groups, as indicated by the red bars. However, SMT's gains come at the cost of even greater skewness in the average utility distribution across groups.

To summarize, all three mechanisms show promising improvements in overall user welfare, but their nature of gains differs, introducing considerations for the platform. When condition allows, HMT is the top choice due to its strong performance, stability, and fairness. For platforms that prioritize fairness and stability, UIR is also a viable option. However, SMT, despite improving overall welfare, may suffer from potential drawbacks such as instability and fairness issues. In-depth analysis of the merits and limitations of these mechanisms remains a topic for future research.

The results in the MovieLens environment align with the insights from the synthetic environment (refer to Figure 3). However, it's worth noting that the trends in learned group weights and realized group utilities do not always align with group size, which is expected in real-world data where unattended user groups may not necessarily have small sizes. Nevertheless, our proposed mechanisms continue to improve overall welfare by identifying and prioritizing these groups.

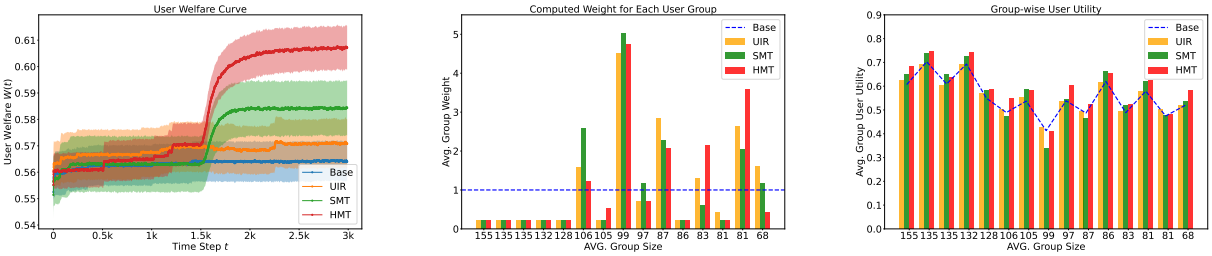
5.2 Online Experiments

We conducted online evaluations on Instagram Reels (IG), one of the world's leading short-video content creation and recommendation platforms (referred to as IG hereafter), spanning over 3 weeks. We observe that the platform's intervention can indeed influence creator behavior because, on average, there is a positive correlation between the delivery volume and content creation volume for each topic. (The Pearson correlation is 0.2, and there are hundreds of



(a) User welfare evolving curve. **(b) Avg. group weight at last iteration.** **(c) Avg. group-wise user utility.**

Figure 2: Performance of UIR, SMT and HMT on synthetic dataset against the no-intervention baseline. Results are averaged over 10 independently sampled synthetic environments including one-sigma error bars. x-axis: group sizes divided by 10.



(a) User welfare evolving curve. **(b) Avg. group weight at last iteration.** **(c) Avg. group-wise user utility.**

Figure 3: Performance of UIR, SMT and HMT on MovieLens-1m dataset against the no-intervention baseline. Results are averaged over 10 independent simulations including 0.2-sigma error bars.

topics in total). In this experiment, we employed the "like-through-rate" (LTR) as the user utility function. LTR is calculated as the ratio of total likes to the number of impressions of a specific short video. We opted for LTR as the chosen metric because it not only serves as a reflection of user satisfaction but also offers a straightforward and easily interpretable signal for content creators to assess their content's perceived quality. The selection of the HMT mechanism for testing was deliberate, driven not only by its strong performance against the baseline and other mechanisms in our offline experiments, but also due to its ease of integration into production: HMT solely requires changing the number of candidate content retrieved for different users within the deployed relevance-based ranking model.

5.2.1 Experiment Setups. We list the experiment setups below.

User clustering: We utilized explicit user characteristics such as demographics including country and gender and their level of activeness including video consumption volume and watch time. This approach led to the creation of over 10,000 user groups and we retained groups that had a sufficient number of users, resulting in hundreds of user groups.

Cluster weight update: We implemented a daily weight updating cadence. Each day, we assessed the satisfaction of every user group by calculating the relative change of LTR over its average in the previous two days. Subsequently, we recalculated the user weights in accordance with the method outlined in Algorithm 1.

A/B test configurations: To evaluate changes in both user and creator behavior, we employed a symmetric A/B test setup on IG. This symmetric A/B test consisted of an experiment arm and a control arm to measure performance. At the beginning, we randomly pair 3% creators with 3% users from the entire platform for each arm. Under this setup, users within each arm exclusively received content created by creators within the same arm, and content created by these creators was exclusively exposed to users within the same arm throughout the testing period. This stringent separation prevents any cross-group treatment leakage and maintains a closed feedback loop within each arm. In our online experiment, we ran these two arms for a duration of 3 weeks: a control arm adhering to the existing production setup and a test arm where we applied our proposed mechanism, HMT.

HMT specifics: We implemented HMT during the cold start content retrieval phase, which pertains to content created within a few days and has not yet garnered a predefined number of impressions. Specifically, within the IG's production pipeline, we integrated an audience matching stage to retrieve cold start content. During this stage, content is exclusively delivered to the most suitable user candidates based on relevance scores generated by a pre-trained model. In the existing production setup, a fixed relevance score percentile of 99% is uniformly applied to all users. This means that every user is only matched with the top 1% of cold-start content in terms of relevance scores to ensure a high level of personalization. When tuning the percentile, we typically observe a trade-off between overall user satisfaction and the volume of cold-start content. In our experiment, we leveraged HMT to intelligently adjust this threshold for different user groups, anticipating improvements in both of thees metrics. Consequently, user groups with higher weights were granted a higher chance to be selected by content creators, while those with lower weights were deprioritized. The mapping from the group weight w to the percentile of retrieved

Table 1: Mapping g in HMT

Weight	< 1.0	< 1.19	< 1.79	< 2.13	< 2.36	< 2.68	≥ 2.68
Percentile	0.99	0.95	0.90	0.85	0.75	0.7	0.1

Table 2: Gains per User Group

User Groups	1-5	6-20	21-74	75+	TOTAL
LTR	+0.43%	+1.40%	+0.75%	+1.36%	**+1.13**%
Impression	+2.64%	+0.62%	+1.42%	+0.11%	**+0.76**%

cold start content proportion was designed as a piece-wise constant function, with details specified in Table 1.

5.2.2 Results. Positive results were obtained in three key aspects.
User-side engagement: The core utility metric LTR increased by 1.13% and the total impression number of cold-start content increased by 0.76%, leading to a 3.7% increase in impressions for fresh content created within 2 hours. These improvements are statistically significant and demonstrate increased user welfare while enhancing the freshness and diversity of content. The gains in both user satisfaction and the volume of cold-start content indicate that HMT influenced many creators to produce more targeted content that benefits niche user groups. Table 2 provides a breakdown of performance improvement per user group. We indexed all groups in descending order by their sizes and divided them into four columns, with each column constituting approximately 25% of the total traffic during the experiment period. As shown, smaller groups enjoyed a higher gain in terms of LTR, which echoes the observations in offline results. The gain in cold-start content impression volume shows an opposite trend. This is because the absolute number of cold-start impressions for larger user groups was smaller as the distribution of relevance scores in this group was more skewed, resulting in a larger relative gain in this metric.
Content diversity: The average number of consumed topics per user during the experimental period increased by 0.71%, and this increase is also statistically significant.
Creator-side engagement: For popular creators (those with more than 1000 followers), the number of daily active users (Creator DAU) increased by an average of 0.17%, while for the remaining creators, the gain is 0.06%. Additionally, there is a promising increasing trend in Creator DAU for popular creators over the three weeks of the experiment: the increases over the first, second, and third weeks are -0.2%, 0.24%, and 0.48%. This suggests that the three-week duration of the experiment may have been too short to influence the majority of creators to respond accordingly, and more time may be needed to fully observe the positive feedback from creators.

6 CONCLUSION

In this study, we tackle the user welfare optimization challenge faced by online content recommendation platforms through the lens of mechanism design. We identified myopic strategy updates among creators caused by their limited information access as the culprit of sub-optimal welfare and introduced platform interventions to address this issue. Our three proposed mechanisms, based on adaptive user importance reweighting, enable platforms to convey global user preference information, reshape creators' perceived utilities, and influence their behaviors. Empirical experiments in both offline and online environments demonstrated the effectiveness of our approach, highlighting its potential for practical impact.

For future work, there remains an intriguing need for a comprehensive understanding of the merits and limitations of UIR, SMT, and HMT to aid practitioners in selecting the most suitable mechanism for real-world applications. It is also important to address practical constraints when applying the developed mechanisms. For instance, can we find ways to jointly optimize user welfare and platform costs? Can the mechanism explicitly ensure fairness on the user side and producer side? Deeper insights into these questions hold the potential to greatly impact the rapidly evolving online content landscape and industry practices.

ACKNOWLEDGEMENT

This work is supported in part by the AI2050 program at Schmidt Sciences (Grant G-24-66104), Army Research Office Award W911NF-23-1-0030, ONR Award N00014-23-1-2802 and NSF Award CCF-2303372.

REFERENCES

[1] Sanjeev Arora, Elad Hazan, and Satyen Kale. 2012. The multiplicative weights update method: a meta-algorithm and applications. *Theory of computing* 8, 1 (2012), 121–164.
[2] E Veronica Belmega, Panayotis Mertikopoulos, Romain Negrel, and Luca Sanguinetti. 2018. Online convex optimization and no-regret learning: Algorithms, guarantees and applications. *arXiv preprint arXiv:1804.04529* (2018).
[3] Omer Ben-Porat and Moshe Tennenholtz. 2017. Shapley facility location games. In *International Conference on Web and Internet Economics*. Springer, 58–73.
[4] 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).
[5] Erdem Biyik, Fan Yao, Yinlam Chow, Alex Haig, Chih-wei Hsu, Mohammad Ghavamzadeh, and Craig Boutilier. 2023. Preference Elicitation with Soft Attributes in Interactive Recommendation. *arXiv preprint arXiv:2311.02085* (2023).
[6] Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. 2013. Recommender systems survey. *Knowledge-based systems* 46 (2013), 109–132.
[7] Craig Boutilier, Martin Mladenov, and Guy Tennenholtz. 2023. Modeling Recommender Ecosystems: Research Challenges at the Intersection of Mechanism Design, Reinforcement Learning and Generative Models. *arXiv preprint arXiv:2309.06375* (2023).
[8] Mario Bravo, David Leslie, and Panayotis Mertikopoulos. 2018. Bandit learning in concave N-person games. *Advances in Neural Information Processing Systems* 31 (2018).
[9] Sarah Dean, Evan Dong, Meena Jagadeesan, and Liu Leqi. 2024. Recommender Systems as Dynamical Systems: Interactions with Viewers and Creators. In *Workshop on Recommendation Ecosystems: Modeling, Optimization and Incentive Design*.
[10] Sarah Dean and Jamie Morgenstern. 2022. Preference dynamics under personalized recommendations. In *Proceedings of the 23rd ACM Conference on Economics and Computation*. 795–816.
[11] Jicong Fan and Jieyu Cheng. 2018. Matrix completion by deep matrix factorization. *Neural Networks* 98 (2018), 34–41.
[12] Daniel Fleder and Kartik Hosanagar. 2009. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management science* 55, 5 (2009), 697–712.
[13] Angela Glotfelter. 2019. Algorithmic circulation: how content creators navigate the effects of algorithms on their work. *Computers and composition* 54 (2019), 102521.
[14] F Maxwell Harper and Joseph A Konstan. 2015. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)* 5, 4 (2015), 1–19.
[15] Thomas Hodgson. 2021. Spotify and the democratisation of music. *Popular Music* 40, 1 (2021), 1–17.
[16] Mattias Holmbom. 2015. The YouTuber: A qualitative study of popular content creators.

[17] Jiri Hron, Karl Krauth, Michael I Jordan, Niki Kilbertus, and Sarah Dean. 2022. Modeling content creator incentives on algorithm-curated platforms. *arXiv preprint arXiv:2206.13102* (2022).
[18] Xinyan Hu, Meena Jagadeesan, Michael I Jordan, and Jacob Steinhard. 2023. Incentivizing High-Quality Content in Online Recommender Systems. *arXiv preprint arXiv:2306.07479* (2023).
[19] Nicole Immorlica, Meena Jagadeesan, and Brendan Lucier. 2024. Clickbait vs. Quality: How Engagement-Based Optimization Shapes the Content Landscape in Online Platforms. In *Proceedings of the ACM Web Conference 2024.*
[20] Meena Jagadeesan, Nikhil Garg, and Jacob Steinhardt. 2022. Supply-Side Equilibria in Recommender Systems. *arXiv preprint arXiv:2206.13489* (2022).
[21] Hanna Kajander. 2019. Challenges of a Content Creator in the Era of Digital Marketing. (2019).
[22] Steven George Krantz and Harold R Parks. 2002. *The implicit function theorem: history, theory, and applications.* Springer Science & Business Media.
[23] 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.
[24] Vaibhavi Nandagiri and Leena Philip. 2018. Impact of influencers from Instagram and YouTube on their followers. *International Journal of Multidisciplinary Research and Modern Education* 4, 1 (2018), 61–65.
[25] John F Nash Jr. 1950. Equilibrium points in n-person games. *Proceedings of the national academy of sciences* 36, 1 (1950), 48–49.
[26] 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).
[27] Kun Qian and Sanjay Jain. 2022. Digital Content Creation: An Analysis of the Impact of Recommendation Systems. *Available at SSRN 4311562* (2022).
[28] J Ben Rosen. 1965. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica: Journal of the Econometric Society* (1965), 520–534.
[29] 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).
[30] Fan Yao, Chuanhao Li, Denis Nekipelov, Hongning Wang, and Haifeng Xu. 2022. Learning from a learning user for optimal recommendations. In *International Conference on Machine Learning*. PMLR, 25382–25406.
[31] Fan Yao, Chuanhao Li, Denis Nekipelov, Hongning Wang, and Haifeng Xu. 2022. Learning the optimal recommendation from explorative users. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 9457–9465.
[32] 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.
[33] Fan Yao, Chuanhao Li, Denis Nekipelov, Hongning Wang, and Haifeng Xu. 2024. Human vs. Generative AI in Content Creation Competition: Symbiosis or Conflict? *arXiv preprint arXiv:2402.15467* (2024).
[34] Fan Yao, Chuanhao Li, Karthik Abinav Sankararaman, Yiming Liao, Yan Zhu, Qifan Wang, Hongning Wang, and Haifeng Xu. 2023. Rethinking Incentives in Recommender Systems: Are Monotone Rewards Always Beneficial? *arXiv preprint arXiv:2306.07893* (2023).
[35] 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.
[36] Banghua Zhu, Sai Praneeth Karimireddy, Jiantao Jiao, and Michael I Jordan. 2023. Online learning in a creator economy. *arXiv preprint arXiv:2305.11381* (2023).

A DETAILS OF CONTENT CREATORS' STRATEGY UPDATE DYNAMICS

LBR 2 captures the evolution of creators' strategies in a snapshot, and characterizes two fundamental properties of content creation: 1. it relies solely on point estimations of the utility function (Line 3); and 2. it only incurs local changes at each update (Line 4). At each step, a creator who decides to update her strategy would first generate an exploration direction $\boldsymbol{g}_i$ (Line 2); then she would evaluate whether adjusting her strategy in this direction results in a higher utility. If so, she proceeds to update her strategy along $\boldsymbol{g}_i$ in a pace of η; otherwise, she maintains her current strategy.

Algorithm 2 closely emulates real-world scenarios where creators strive to optimize their utilities while having merely black-box access to the utility functions. In practice, finding a clear direction that guarantees improved utility can be a challenging and, at times, unrealistic task. Consequently, we model their strategy evolution as an iterative process of trial and error. By definition, when LBR converges in C^3_{ext}, it must converge to an LNE. Our primary interest lies in understanding how the platform can devise a dynamic rewarding or matching principle that maximizes cumulative user welfare within a given time period.

Algorithm 2 (**LBR**) Local Better Response update at time step t

1: **Input:** Learning rate η, an C^3_{ext} instance including utility functions and strategy sets $(u_i(\boldsymbol{s}), \mathcal{S}_i)$ of creator i, the joint strategy profile $\boldsymbol{s}^{(t)} = (\boldsymbol{s}_1^{(t)}, \cdots, \boldsymbol{s}_n^{(t)})$ at the current step t.
2: Generate a random direction $\boldsymbol{g}_i \in \mathbb{S}^d$.
3: **if** $u_i(\boldsymbol{s}_i^{(t)} + \eta \boldsymbol{g}_i, \boldsymbol{s}_{-i}^{(t)}) \geq u_i(\boldsymbol{s}^{(t)})$ **then**
4: $\quad \boldsymbol{s}_i^{(t+\frac{1}{2})} = \boldsymbol{s}_i^{(t)} + \eta \boldsymbol{g}_i$.
5: $\quad$ Find $\boldsymbol{s}_i^{(t+1)}$ as the projection of $\boldsymbol{s}_i^{(t+\frac{1}{2})}$ in $\mathcal{S}_i$.
6: **else**
7: $\quad \boldsymbol{s}_i^{(t+1)} = \boldsymbol{s}_i^{(t)}$

B IMPLEMENTATION DETAILS OF UIR, SMT AND HMT MECHANISMS

The following sub-routine, denoted as Algorithm 3, outlines how the platform deploys the weights obtained from Line 8, Algorithm 1 as an intervention mechanism in Line 4. In Algorithm 3, the weight vector $\boldsymbol{w}$ is directly employed to modify the reward or payment associated with each creator-user interaction.

Algorithm 3 UIR Intervention

Input: Default recall capacity K, matching temperature β.
for each user request $\boldsymbol{x}$ **do**
Compute the relevance scores $\{\sigma(\boldsymbol{s}_i, \boldsymbol{x})\}_{i=1}^n$.
Retrieve the top-K ranked content $\{s_{l(1)}, \cdots, s_{l(K)}\}$ list based on relevance scores and randomly sample one element according to $\text{Softmax}(\{\beta^{-1}\sigma(\boldsymbol{s}_{l(i)}, \boldsymbol{x})\}_{i=1}^K)$.
For the user's choice $\boldsymbol{s}_i$, adjust creator-i's default reward (payment) from $R(\boldsymbol{s}_i, \boldsymbol{x})$ to $w(\boldsymbol{x})R(\boldsymbol{s}_i, \boldsymbol{x})$.

In the case of SMT or HMT intervention types, the platform requires a function to map $w(\boldsymbol{x})$ to $\beta(\boldsymbol{x})$ or $K(\boldsymbol{x})$. This mapping can be implemented as a piecewise constant function and determined empirically. The specifics of this process are elucidated in Algorithm 4 and 5.

C PROOF OF THEOREM 3.1

We restate Theorem 3.1 as the following with more rigorous characterizations, and then provide its detailed proof.

Theorem C.1. *Any C^3_{ext} game with $K = n$ has a unique pure Nash equilibrium (PNE) if each creator's srtategy set $\mathcal{S}_i$ is convex and $\sigma(\cdot, \boldsymbol{x})$ is twice-differentiable and satisfies*

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\left[\frac{\partial^2\sigma}{\partial \boldsymbol{s}_i^2} + \left(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\right)\left(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\right)^\top\right] \preceq 0, \forall i \in [n]. \tag{15}$$

Algorithm 4 SMT Intervention

Input: Default recall capacity K, matching temperature β, $f : \mathbb{R}_+ \to \mathbb{R}_+$.
for each user request $\boldsymbol{x}$ **do**
Compute the relevance scores $\{\sigma(\boldsymbol{s}_i, \boldsymbol{x})\}_{i=1}^n$.
Retrieve the top-K ranked content $\{s_{l(1)}, \cdots, s_{l(K)}\}$ list based on relevance scores and randomly sample one element according to Softmax$(\{\beta(\boldsymbol{x})^{-1}\sigma(\boldsymbol{s}_{l(i)}, \boldsymbol{x})\}_{i=1}^K)$, where $\beta(\boldsymbol{x}) = f(w(\boldsymbol{x}))$.

Algorithm 5 HMT Intervention

Input: Default recall capacity K, matching temperature β, $g : \mathbb{R}_+ \to \mathbb{N}_+$.
for each user request $\boldsymbol{x}$ **do**
Compute the relevance scores $\{\sigma(\boldsymbol{s}_i, \boldsymbol{x})\}_{i=1}^n$.
Retrieve the top-$K(\boldsymbol{x})$ ranked content $\{s_{l(1)}, \cdots, s_{l(K(\boldsymbol{x}))}\}$ list based on relevance scores and randomly sample one element according to Softmax$(\{\beta^{-1}\sigma(\boldsymbol{s}_{l(i)}, \boldsymbol{x})\}_{i=1}^K(\boldsymbol{x}))$, where $K(\boldsymbol{x}) = g(w(\boldsymbol{x}))$.

Proof. We prove that under the proposed conditions, the C^3_{ext} is a strictly monotone game [28] and thus possesses a unique PNE. According to Appendix A in [8], a sufficient condition that establishes strictly monotonicity for any n-person game $\mathcal{G}$ is convex action sets and a negative definite Hessian $[H_{ij}^{\mathcal{G}}]$ of $\mathcal{G}$, which is defined as

$$H_{ij}(\boldsymbol{s}) = \frac{1}{2}\nabla_j\nabla_i u_i(\boldsymbol{s}) + \frac{1}{2}\nabla_i\nabla_j u_j(\boldsymbol{s})^\top.$$

For C^3_{ext} game, the convexity of strategy sets are satisfied. Next we prove the property of the game's Hessian matrix with associated utility function

$$u_i(\boldsymbol{s}) = \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\left[\frac{\exp(\sigma(\boldsymbol{s}_i, \boldsymbol{x}))}{\sum_{l=1}^n \exp(\sigma(\boldsymbol{s}_l, \boldsymbol{x}))}\right]. \tag{16}$$

Without loss of generality, let $\beta = 1$. Denote $A_i = \exp(\sigma(\boldsymbol{s}_i, \boldsymbol{x}))$, $M = A_1 + \cdots + A_n$, we have

$$\begin{aligned}
H_{ii} &= -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big\{\Big[-\frac{\partial^2\sigma}{\partial \boldsymbol{s}_i^2} - \Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)^\top\Big]A_i(M - A_i)\cdot\frac{1}{M^2} \\
&\qquad + \frac{2}{M}\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)^\top A_i^2(M - A_i)\cdot\frac{1}{M^2}\Big\} \\
&= -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big\{\Big[-\frac{\partial^2\sigma}{\partial \boldsymbol{s}_i^2} - \Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)^\top\Big(1 - \frac{A_i}{M}\Big)\Big]\cdot A_i(M - A_i)\frac{1}{M^2}\Big\} \\
&\quad - \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big\{\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)^\top A_i^2(M - A_i)\cdot\frac{1}{M^3}\Big\} \\
&\triangleq -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big[H_{ii}^{(0)}(\boldsymbol{s}, \boldsymbol{x})\Big] - \mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big[H_{ii}^{(1)}(\boldsymbol{s}, \boldsymbol{x})\frac{1}{M^3}\Big].
\end{aligned}$$

$$\begin{aligned}
H_{ij} &= -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big\{\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_j}\Big)^\top A_iA_j(M - A_i - A_j)\cdot\frac{1}{M^3}\Big\} \\
&\triangleq -\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\Big[H_{ij}^{(1)}(\boldsymbol{s}, \boldsymbol{x})\frac{1}{M^3}\Big].
\end{aligned}$$

Next we show that for any $\boldsymbol{x}$ and $\boldsymbol{s}$, the block matrix $[H_{ij}^{(1)}]$ is always positive semi-definite (PSD). For simplicity, let

$$\begin{aligned}
&\boldsymbol{y}_i = A_i\frac{\partial\sigma}{\partial \boldsymbol{s}_i} \in \mathbb{R}^{d\times 1}, \boldsymbol{y} = [\boldsymbol{y}_1; \ldots; \boldsymbol{y}_n] \in \mathbb{R}^{dn\times 1}, \\
&\boldsymbol{z} = [A_1\boldsymbol{y}_1; \ldots; A_n\boldsymbol{y}_n] \in \mathbb{R}^{dn\times 1},
\end{aligned}$$

we obtain

$$\begin{aligned}
&[H_{ij}^{(1)}] = \\
&\begin{bmatrix}
\boldsymbol{y}_1\boldsymbol{y}_1^\top(M - A_1) & \boldsymbol{y}_1\boldsymbol{y}_2^\top(M - A_1 - A_2) & \cdots & \boldsymbol{y}_1\boldsymbol{y}_n^\top(M - A_1 - A_n) \\
\boldsymbol{y}_2\boldsymbol{y}_1^\top(M - A_2 - A_1) & \boldsymbol{y}_2\boldsymbol{y}_2^\top(M - A_2) & \cdots & \boldsymbol{y}_2\boldsymbol{y}_n^\top(M - A_2 - A_n) \\
\vdots & \vdots & \ddots & \vdots \\
\boldsymbol{y}_n\boldsymbol{y}_1^\top(M - A_n - A_1) & \boldsymbol{y}_n\boldsymbol{y}_2^\top(M - A_n - A_2) & \cdots & \boldsymbol{y}_n\boldsymbol{y}_n^\top(M - A_n)
\end{bmatrix} \\
&= M\boldsymbol{y}\boldsymbol{y}^\top - \boldsymbol{y}\boldsymbol{z}^\top - \boldsymbol{z}\boldsymbol{y}^\top + \text{diag}(A_1\boldsymbol{y}_1\boldsymbol{y}_1^\top, \ldots, A_n\boldsymbol{y}_n\boldsymbol{y}_n^\top) \\
&= \frac{1}{M}\cdot(M\boldsymbol{y} - \boldsymbol{z})(M\boldsymbol{y} - \boldsymbol{z})^\top + \text{diag}(A_1\boldsymbol{y}_1\boldsymbol{y}_1^\top, \ldots, A_n\boldsymbol{y}_n\boldsymbol{y}_n^\top) - \frac{1}{M}\boldsymbol{z}\boldsymbol{z}^\top \\
&> \text{diag}(A_1\boldsymbol{y}_1\boldsymbol{y}_1^\top, \ldots, A_n\boldsymbol{y}_n\boldsymbol{y}_n^\top) - \frac{1}{M}\boldsymbol{z}\boldsymbol{z}^\top.
\end{aligned}$$

Therefore, it suffices to prove that the matrix

$$\tilde{H} = M\text{diag}(A_1\boldsymbol{y}_1\boldsymbol{y}_1^\top, \ldots, A_n\boldsymbol{y}_n\boldsymbol{y}_n^\top) - \boldsymbol{z}\boldsymbol{z}^\top$$

is PSD. For any $\boldsymbol{t} = [\boldsymbol{t}_1; \cdots; \boldsymbol{t}_n] \in \mathbb{R}^{dn\times 1}$ where $\boldsymbol{t}_i \in \mathbb{R}^d$, we can verify that

$$\begin{aligned}
\boldsymbol{t}^\top\tilde{H}\boldsymbol{t} &= M\sum_{i=1}^n A_i\boldsymbol{t}_i^\top\boldsymbol{y}_i\boldsymbol{y}_i^\top\boldsymbol{t}_i - \boldsymbol{t}^\top\boldsymbol{z}\boldsymbol{z}^\top\boldsymbol{t} \\
&= \sum_{i=1}^n A_i\sum_{i=1}^n A_i\boldsymbol{t}_i^\top\boldsymbol{y}_i\boldsymbol{y}_i^\top\boldsymbol{t}_i - \boldsymbol{t}^\top\boldsymbol{z}\boldsymbol{z}^\top\boldsymbol{t} \\
&= \sum_{1\le i<j\le n} A_iA_j(\boldsymbol{y}_i^\top\boldsymbol{t}_i - \boldsymbol{y}_j^\top\boldsymbol{t}_j)^2 \ge 0.
\end{aligned}$$

Therefore, the block matrix $[H_{ij}^{(1)}]$ is always PSD for any $\boldsymbol{x}$ and $\boldsymbol{s}$. A sufficient condition for $[H_{ij}^{\mathcal{G}}]$ to be negative definite is thus $H_{ii}^{(0)}$ being positive definite (PD), i.e., $H_{ii}^{(0)}(\boldsymbol{s}, \boldsymbol{x}) > 0, \forall \boldsymbol{s}, \boldsymbol{x}$. It remains to show that

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\left[\Big[-\frac{\partial^2\sigma}{\partial \boldsymbol{s}_i^2} - \Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)^\top\Big(1 - \frac{A_i}{M}\Big)\Big]\cdot A_i(M - A_i)\frac{1}{M^2}\right] > 0. \tag{17}$$

And a sufficient condition for Eq (17) to hold is

$$\mathbb{E}_{\boldsymbol{x}\sim\mathcal{X}}\left[-\frac{\partial^2\sigma}{\partial \boldsymbol{s}_i^2} - \Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)\Big(\frac{\partial\sigma}{\partial \boldsymbol{s}_i}\Big)^\top\right] \ge 0, \tag{18}$$

which completes the proof. □

D PROOF OF THEOREM 4.1

Proof. Since the utility functions of C^3_{ext} are twice differentiable, any LNE $\boldsymbol{s}$ of C^3_{ext} satisfies the following definition

$$\boldsymbol{s}_i = \arg\max_{\boldsymbol{z}_i\in B(\boldsymbol{s}_i,\delta)} u_i(\boldsymbol{z}_i, \boldsymbol{s}_{-i}; \boldsymbol{w}) \tag{19}$$

must also satisfy the first-order condition $\frac{\partial u_i}{\partial s_i}\Big|_{\boldsymbol{s}=(s_i,\boldsymbol{s}_{-i})} = 0$. If we let

$$F(\boldsymbol{s},\boldsymbol{w}) = \left(\frac{\partial u_1(\boldsymbol{s};\boldsymbol{w})}{\partial \boldsymbol{s}_1}, \cdots, \frac{\partial u_n(\boldsymbol{s};\boldsymbol{w})}{\partial \boldsymbol{s}_n}\right) : \mathbb{R}^{dn} \times \mathbb{R}^m_{\geq 0} \to \mathbb{R}^{dn} \quad (20)$$

be a vector-valued function, the constraint (12) can be rewritten into

$$F(\boldsymbol{s}^*(\boldsymbol{w}),\boldsymbol{w}) = 0. \quad (21)$$

From the implicit function theorem [22], the derivative of $\boldsymbol{s}^*$ w.r.t. $\boldsymbol{w}$ can be written as

$$\frac{d\boldsymbol{s}}{d\boldsymbol{w}} = -\left(\frac{\partial F}{\partial \boldsymbol{s}}\right)^{-1} \cdot \frac{\partial F}{\partial \boldsymbol{w}},$$

where $\left[\frac{\partial F}{\partial \boldsymbol{s}}\right]_{nd\times nd}, \left[\frac{\partial F}{\partial \boldsymbol{w}}\right]_{nd\times m}$ are the Jacobian matrices, and

$$\frac{dW}{d\boldsymbol{w}} = \frac{dW}{d\boldsymbol{s}} \cdot \frac{d\boldsymbol{s}}{d\boldsymbol{w}} = -\frac{dW}{d\boldsymbol{s}} \cdot \left(\frac{\partial F}{\partial \boldsymbol{s}}\right)^{-1} \cdot \frac{\partial F}{\partial \boldsymbol{w}}, \quad (22)$$

where $\left(\frac{dW}{d\boldsymbol{s}}\right)_{1\times nd}$ is the partial derivative of W w.r.t. $\boldsymbol{s}$.

Since $w_j \geq 0$, we apply a change of variable and denote each w_j as e^{w_j} instead. Next we calculate each term of the RHS of (22) to obtain an estimation of the gradient of our objective welfare function W to the user weight vector $\boldsymbol{w}$. Without loss of generality we let the user distribution $\mathcal{X}$ be a uniform distribution on unit basis $\{\mathbf{e}_1, \cdots, \mathbf{e}_d\}$ and $m = d$. The utility functions given in Eq (6) and the user welfare function read

$$W(\boldsymbol{s}) = \frac{1}{m}\sum_{j=1}^{d}\sum_{i=1}^{n} \boldsymbol{s}_i^\top \boldsymbol{x}_j \cdot \frac{\exp[\beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_j]}{\sum_{k=1}^n \exp[\beta^{-1}\boldsymbol{s}_k^\top \boldsymbol{x}_j]}. \quad (23)$$

$$u_i(\boldsymbol{s}_i, \boldsymbol{s}_{-i}) = \frac{1}{m}\sum_{j=1}^{d} e^{w_j} \cdot \frac{\exp[\beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_j]}{\sum_{k=1}^n \exp[\beta^{-1}\boldsymbol{s}_k^\top \boldsymbol{x}_j]}, i \in [K]. \quad (24)$$

If we denote $A_{ij} = \exp[\beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_j], M_j = \sum_{k=1}^n \exp[\beta^{-1}\boldsymbol{s}_k^\top \boldsymbol{x}_j]$, then $\frac{A_{ij}}{M_j} = P_i(\boldsymbol{s}, \boldsymbol{x}_j)$ is exactly the probability of matching content $\boldsymbol{s}_i$ to $\boldsymbol{x}_j$. Given the assumption that n is sufficiently large, we have $\frac{A_{ij}}{M_j} = o(1)$ is sufficiently small for any i and therefore we ignore the high-order infiintesimal terms such as $\frac{A_{ij}^2}{M_j^2}, \frac{A_{kj}A_{ij}}{M_j^2}$ in the following derivation.

$$\begin{aligned}\frac{dW}{d\boldsymbol{s}_i} =& \frac{1}{m}\sum_{j=1}^{d} \boldsymbol{x}_j \left[\frac{A_{ij}}{M_j} + \beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_j \left(\frac{A_{ij}}{M_j} - \frac{A_{ij}^2}{M_j^2}\right)\right] \\ & - \frac{1}{m}\sum_{j=1}^{d} \boldsymbol{x}_j \left[\sum_{k\neq i} \beta^{-1}\boldsymbol{s}_k^\top \boldsymbol{x}_j \frac{A_{kj}A_{ij}}{M_j^2}\right] \\ \approx& \frac{1}{m}\sum_{j=1}^{d} \boldsymbol{x}_j \frac{A_{ij}}{M_j}\left(1 + \beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_j\right), i \in [n], \end{aligned} \quad (25)$$

where $\bar{\pi}(\boldsymbol{x}_j) \triangleq \sum_{k=1}^n \boldsymbol{s}_k^\top \boldsymbol{x}_j \frac{A_{kj}}{M_j}$.

Next we calculate each term in the RHS of Eq (22). The i-th block of $F(\boldsymbol{s},\boldsymbol{w})$ is a d-dimensional vector given by

$$\begin{aligned} F(\boldsymbol{s},\boldsymbol{w})_i =& \frac{1}{m}\sum_{j=1}^{d} \boldsymbol{x}_j \left[\beta^{-1} e^{w_j}\left(\frac{A_{ij}}{M_j} - \frac{A_{ij}^2}{M_j^2}\right)\right] \\ \approx& \frac{1}{m}\sum_{j=1}^{d} \boldsymbol{x}_j \frac{A_{ij}}{M_j}\beta^{-1} e^{w_j}, i \in [n], \end{aligned} \quad (26)$$

the (i,j)-th block of $\frac{\partial F}{\partial \boldsymbol{w}}$ is a d-dimensional vector given by

$$\begin{aligned}\left[\frac{\partial F}{\partial \boldsymbol{w}}\right]_{ij} =& \frac{1}{m}\boldsymbol{x}_j \beta^{-1} e^{w_j}\left(\frac{A_{ij}}{M_j} - \frac{A_{ij}^2}{M_j^2}\right) \\ \approx& \frac{1}{m}\boldsymbol{x}_j \beta^{-1}\frac{A_{ij}}{M_j} e^{w_j}, i \in [n], j \in [d]. \end{aligned} \quad (27)$$

Since $\{\boldsymbol{x}_i\}_{i=1}^n$ are orthogonal basis, the non-diagonal blocks of matrix $\frac{\partial F}{\partial \boldsymbol{s}}$ are all zero matrices and the i-th diagonal block of matrix $\frac{\partial F}{\partial \boldsymbol{s}}$ is given by

$$\begin{aligned}\left[\frac{\partial F}{\partial \boldsymbol{s}}\right]_{ii} =& \frac{1}{m\beta^2}\sum_{j=1}^{d} e^{w_j}\left[\boldsymbol{x}_j \boldsymbol{x}_j^\top \left(\frac{A_{ij}}{M_j} - \frac{3A_{ij}^2}{M_j^2} + \frac{2A_{ij}^3}{M_j^3}\right)\right] \\ \approx& \frac{1}{m\beta^2}\sum_{j=1}^{d} e^{w_j}\left[\boldsymbol{x}_j \boldsymbol{x}_j^\top \frac{A_{ij}}{M_j}\right], i \in [n]. \end{aligned} \quad (28)$$

Therefore, we can derive a approximation of $\frac{dW}{d\boldsymbol{w}}$ as below:

$$\begin{aligned}\frac{dW}{dw_j} =& -\frac{dW}{d\boldsymbol{s}} \cdot \left(\frac{\partial F}{\partial \boldsymbol{s}}\right)^{-1} \cdot \left(\frac{\partial F}{\partial \boldsymbol{w}}\right)_j \\ \approx& -\sum_{i=1}^{n}\Bigg\{\frac{1}{m}\sum_{k=1}^{d} \boldsymbol{x}_k^\top \frac{A_{ik}}{M_k}\left(1 + \beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_k\right) \\ & \cdot m\beta^2 \mathrm{diag}^{-1}(e^{w_1}A_{i1}/M_1, \cdots, e^{w_L}A_{id}/M_d) \cdot \frac{1}{m}\boldsymbol{x}_j \beta^{-1}\frac{A_{ij}}{M_j} e^{w_j}\Bigg\} \\ =& -\frac{\beta^2}{m}\sum_{i=1}^{n} e^{-w_j}\left(1 + \beta^{-1}\boldsymbol{s}_i^\top \boldsymbol{x}_j\right)\beta^{-1}\frac{A_{ij}}{M_j} e^{w_j} \\ \approx& -\frac{1}{m}\sum_{i=1}^{n} \boldsymbol{s}_i^\top \boldsymbol{x}_j \frac{A_{ij}}{M_j} \quad (29) \\ =& -\frac{1}{m}\sum_{i=1}^{n} \pi(\boldsymbol{s}_i, \boldsymbol{x}_j) P_i(\boldsymbol{s}, \boldsymbol{x}_j) \\ =& -\frac{1}{m}\mathbb{E}_{\boldsymbol{s}}[\pi(\boldsymbol{x}_j)], \quad (30) \end{aligned}$$

where (29) holds because $\beta^{-1} >> 1$.

Therefore, Eq (30) suggests that the following update rule

$$e^{w'_j} = e^{w_j} \cdot e^{-\eta\bar{\pi}(\boldsymbol{x}_j)} \quad (31)$$

aligns with the gradient direction of $W(\boldsymbol{w})$, which yields Eq (13). □