Journal of Intelligent Information Systems
https://doi.org/10.1007/s10844-026-01025-y

RESEARCH

Dynamic feedback loops in recommender systems: Analyzing fairness, popularity bias, and user group disparities

Yildiz Zoralioglu[1] · Emre Yalcin[2]

Received: 4 February 2025 / Revised: 29 December 2025 / Accepted: 5 January 2026

Abstract

Ensuring equitable treatment of different user groups in recommender systems is a key challenge, and the issue of fairness has been widely explored in the literature. However, understanding fairness within a robust feedback loop, as it occurs in real-world settings, remains elusive. This study examines the interplay between popularity bias, calibration, accuracy, and beyond-accuracy performance of recommender systems using a novel dynamic feedback loop framework. The framework models iterative interactions between recommendation algorithms and user profiles, enabling the analysis of calibration, accuracy, and beyond-accuracy measures across user groups with varying preferences on popular items, i.e., *Popular-*, *Diverse-*, and *Niche-focused*. Empirical evaluations conducted on two benchmark datasets using three collaborative filtering algorithms reveal distinct disparities in how feedback loops affect different user groups. *Niche-focused* users, despite being the most active and information-rich, experience the steepest deterioration in system alignment over time, losing much of their initial calibration, long-tail exposure, and diversity advantages, along with proportional declines in accuracy. These results show that feedback dynamics progressively misalign the system with its most valuable users, making them the most disadvantaged over time. *Popular-focused* users remain most aligned with algorithmic tendencies, achieving steady accuracy gains but remaining confined to narrow, popularity-driven content with little to no long-tail exposure. Meanwhile, *Diverse-focused* users, initially balanced between popular and niche preferences, undergo gradual calibration drift and consistent reductions in diversity and long-tail representation, gradually converging toward recommendation patterns similar to *Popular-focused* users. Overall, the results demonstrate that feedback loops magnify structural inequalities, reinforcing popularity bias while reducing diversity and personalization across all user groups.

Keywords Recommender systems · Collaborative filtering · Fairness · Popularity bias · Feedback loop

Emre Yalcin contributed equally to this work.

Extended author information available on the last page of the article

Published online: 14 January 2026

1 Introduction

Recommender systems (RSs) have become a cornerstone of artificial intelligence by addressing the urgent challenge of information overload through personalized content delivery, tailored to individual user needs (Wei and Chen, 2025; Vercoutere et al., 2025). They find application in domains such as e-commerce, digital media, education, and healthcare, enhancing user experiences with precise, context-aware recommendations (Suhaim and Berri, 2021; Alizadeh Noughabi et al., 2025). Leading platforms, including Netflix, Spotify, and Amazon, illustrate the transformative potential of RSs by analyzing user behaviors to provide customized suggestions for movies, music, and products (Bobadilla et al., 2023).

Within the domain of RSs, collaborative filtering (CF) has emerged as a prevalent and impactful methodology. CF derives insights from user-item interactions by analyzing patterns in user behavior (user-based CF) or identifying shared attributes among items (item-based CF), enabling the prediction of unobserved user preferences. Bogers and Van Den Bosch (2009). Despite its effectiveness, CF-based algorithms grapple with inherent challenges, notably the cold-start problem, where limited data on new users or items hinders accurate recommendations, and data sparsity, which stems from the generally sparse interaction data in large-scale datasets (Guan et al., 2024).

Along with these challenges, RSs also face popularity bias. This happens when popular items keep getting recommended, while niche or less-known items are ignored (Yalcin and Bilge, 2022a). Bias often arises because a small number of items receive most user interactions, while many items remain underrepresented (Klimashevskaia et al., 2024). As a result, the user-item rating matrix that CF relies on becomes imbalanced, causing algorithms trained on this data to favor popular items and ignore less-rated options (Lin et al., 2022). This effect can form a self-reinforcing cycle: once certain items become more visible, they attract even more interactions, boosting their popularity further (Chen et al., 2023). Popularity bias also leads to calibration issues, where recommendations may not match individual user tastes for popular or niche content, reducing personalization and user satisfaction (Klimashevskaia et al., 2024).

Popularity bias has wide-ranging effects on users, content creators, and service providers. Users find it harder to discover diverse or new content, which reduces the personalization and enjoyment that RSs aim to provide. Content creators and niche products get limited visibility and fewer opportunities to grow, resulting in an uneven playing field where popular items dominate (Sharma et al., 2021). Service providers face lower user satisfaction, engagement, and revenue when recommendations lack variety (Marcuzzo et al., 2022). Popularity bias also harms beyond-accuracy measures such as novelty, coverage, and diversity: recommendations grow repetitive, fewer unique items are shown, and popular items dominate (Li, 2023). These issues not only weaken user experience but also raise fairness concerns, hurting users with niche preferences and content creators with less mainstream products. Because of these concerns, popularity bias has become a major topic in RS research, spurring various methods to reduce its harmful effects.

Recent work also shows how popularity bias affects different user groups. These include protected attributes (like age, gender) (Lesota et al., 2021) and unprotected attributes (like rating patterns, personality) (Yalcin and Bilge, 2022a; Abdollahpour, 2020; Yalcin and Bilge, 2023). One prominent study categorized users based on their interest in item popularity and found that many algorithms put too much focus on popular content compared to what some user groups actually prefer (Abdollahpouri et al., 2019b). *Niche-focused* users, in particular, suffer the most because they receive fewer recommendations that match their preferences, while blockbuster-focused users experience less harm since they naturally enjoy popular items. However, that study only looked at a single round of recommendations in an offline setting, which did not include a feedback loop mechanism. Building on these insights, various strategies have been proposed to make recommendations more calibrated (Abdollahpouri et al., 2021; Gulsoy et al., 2025), balancing user preferences with item fairness. These strategies highlight the need for personalized methods that reflect the unique traits of different users, aiming for a more diverse and fair recommendation environment.

Even though popularity bias and fairness are important, many studies only use static evaluations, focusing on one-time/shot recommendations and their immediate effects. In practice, RSs run in dynamic settings, where user preferences change over multiple recommendation cycles. These cycles create a two-way effect: recommendations influence user actions, which then shape how the recommendation model learns. Although real-time feedback loops could offer deeper insights, offline experiments usually cannot collect user feedback for each cycle because they do not have access to live user interactions, something that only platforms can do. One of the few studies that looks at this dynamic view found that algorithms often make popularity bias worse by repeatedly suggesting already-popular items, pushing user activity toward a small set of top-rated choices (Mansoury et al., 2020). While that work illustrates how the cycle reduces overall item diversity, skews item representation over time, and disproportionately disadvantages certain item categories, it does not propose a user-based classification or analysis to investigate how feedback loops might treat different user segments unfairly.

In this study, we propose a feedback loop framework to explore how recommendation quality changes over time in several areas including accuracy, beyond-accuracy, fairness, and calibration in terms of popularity for users with different levels of interest in popular items. The following summarizes our key contributions:

- We introduce a practical framework for analyzing feedback in RSs, addressing the lack of a common standard for such studies. In each cycle, our framework generates synthetic ratings for users based on two explicitly separated scenarios: *repeat consumption* and *new consumption*. Repeat consumption preserves a user's historical ratings while adding user-specific Gaussian noise to capture natural preference variability, whereas new consumption predictions are produced via a scale-consistent item-based CF method with controlled noise. This design ensures realistic simulation of user behavior and preserves the original rating distribution across cycles.
- We provide an analysis monitoring how rating distributions evolve over time, ensuring that the simulation process does not artificially narrow variance or distort the dataset's statistical structure.
- We track how fairness in recommendations changes for three different user groups: *popularity-*, *diversity-*, and *niche-focused* based on their original tendency toward popular items. We perform a detailed, long-term analysis of recommendation quality, looking at multiple metrics and dimensions, including accuracy, beyond-accuracy measures (i.e., long-tail exposure, diversity, and coverage), fairness and calibration in terms of popularity.

The remainder of this paper is organized as follows: Section 2 reviews related work on popularity bias, fairness, and feedback mechanisms in RSs. Section 3 introduces our proposed simulation environment for modeling iterative feedback loops and provides a detailed description of the methodology. Section 4 outlines the experimental setup, including the employed methodology, utilized datasets, applied CF algorithms, and evaluation metrics. Section 5 presents the results of the experiments and discusses the key insights gained. Section 6 discusses the main limitations of our study. Finally, Section 7 concludes the paper by summarizing the main findings and proposing directions for future research.

2 Related work

Popularity bias in recommender algorithms has emerged as a significant and complex challenge in the research landscape (Boratto et al., 2021; Abdollahpouri et al., 2017). It occurs when algorithms disproportionately favor popular items while neglecting less popular or niche alternatives (Kowald et al., 2020; Jannach et al., 2015), thereby limiting users' access to diverse content, reducing revenue opportunities for content providers, and creating systemic imbalances.

To address these issues, researchers have introduced various methods aimed at mitigating popularity bias and increasing the likelihood of recommending less popular items (Borges and Stefanidis, 2021). Such methods are often categorized as *pre-processing*, *in-processing*, or *post-processing* strategies. *Pre-processing* approaches alter the existing user-item matrix to reduce imbalances (Waris et al., 2024). *In-processing* methods modify the internal mechanisms of recommendation algorithms to improve the representation of niche items. *Post-processing* strategies re-rank recommendations to penalize popular items (Yalcin and Bilge, 2022b). Although these approaches strive to balance accuracy and diversity, they also highlight broader fairness concerns. Overemphasis on popular content limits users' access to niche items, thus disadvantaging users on one hand, and reducing the visibility of less popular items, thus disadvantaging items on the other.

Kowald et al. (2020) show that popularity bias in music recommendations restricts the discovery of niche content and harms user experiences. Abdollahpouri et al. (2019b) further point out that popularity bias produces unfair outcomes for both users and items, ultimately degrading overall recommender performance. Consequently, users who prefer niche items experience higher levels of unfairness compared to those inclined toward popular content. In another related work on fairness, Abdollahpouri and Burke (2019) and Burke (2017) propose a framework that classifies fairness into three dimensions: C-fairness (users interacting with recommendations), P-fairness (content providers), and S-fairness (stakeholders indirectly impacted by recommendations).

The findings observed in our previous work (Yalcin and Bilge, 2023) reveal that individuals with different personality traits are impacted differently by popularity bias; for instance, less extroverted users or those who prefer novelty face greater unfairness in popularity-centric recommendations, despite their substantial contributions to the system. We have also examined how rating profile characteristics, such as profile size, rating deviation, average rating, entropy, and consistency, influence users' vulnerability to popularity bias (Yalcin and Bilge, 2022a). Highly engaged, selective, and hard-to-predict users receive disproportionately unfair, relatively inaccurate, and suboptimal recommendations from a beyond-accuracy perspective, even though they are major system stakeholders. Another study explored fairness in a privacy-preserved environment by classifying users according to their preferred privacy levels (Gulsoy et al., 2023), showing that privacy-sensitive users could receive unbiased, fairer recommendations, especially in terms of diversity, novelty, and catalog coverage, with only a small trade-off in accuracy. Hence, ensuring fair recommendation strategies that explicitly address popularity bias is crucial to promote equitable exposure and maintain user satisfaction.

A recent study introduces the concept of calibration in recommendations (Steck, 2018), where suggested items are expected to align with a user's historical content proportions. For example, if a user assigns 40% of their ratings to romance films and 60% to horror films, the recommendation list should reflect similar ratios. Rather than relying on genre-based distributions, Abdollahpouri et al. (2019b) and Kowald et al. (2020) examine calibration based on item popularity, arguing that recommendations should match a user's genuine interest in either popular or niche content. This user-centric view of popularity bias has been adopted to offer more calibrated suggestions (Abdollahpouri et al., 2021). Meanwhile, Abdollahpouri et al. (2019a) propose the idea of miscalibration, suggesting that if it becomes widespread, the system may be failing to personalize effectively and thus unfairly disadvantaging certain user groups. Fair proportionality, in this sense, involves applying attributes such as labels, categories, genres, or classes to ensure that recommendations align with users' actual preferences. Finally, a recent work introduces *DUoR* method as a novel user-centric strategy to mitigate popularity bias in RSs (Gulsoy et al., 2025). Unlike conventional methods, it measures users' genuine inclination toward popular content by focusing only on items they truly enjoyed and integrates this into an iterative re-ranking process. Experimental results on benchmark datasets show that *DUoR* achieves more calibrated, fair, and diverse recommendations compared to existing debiasing approaches.

Feedback loops represent another critical factor in understanding fairness from a dynamic standpoint. These loops can exacerbate popularity bias over time, as user interactions amplify the focus on already popular items. Mansoury et al. (2020) found that user interactions reinforce algorithmic preferences for popular content, shrinking the recommendation space and lowering diversity. Also, Chen et al. (2016) investigate the distinction between first-time and repeat consumption in RSs. They propose a probabilistic model to capture users' re-consumption tendencies, demonstrating that treating repeated and novel interactions differently can lead to more realistic simulation outcomes and evaluation protocols. Also Boutilier et al. (2024) present a detailed simulation framework that incorporates realistic user behavior models into feedback loop analysis. Their approach captures heterogeneity in user preferences and interaction patterns, demonstrating how nuanced behavioral modeling can yield more robust and generalizable insights into long-term recommendation dynamics. Similarly, Mladenov et al. (2020) introduced RecSim NG, a modular, probabilistic, and differentiable simulation platform designed to evaluate sequential decision-making in RSs. This environment facilitates controlled experimentation with user and item models, enabling reproducible studies of long-term effects such as popularity bias and fairness. Finally, Yao et al. (2023) proposed a simulation-based framework to jointly evaluate popularity bias and fairness under dynamic conditions. They emphasize the importance of incorporating beyond-accuracy metrics into the analysis and show that fairness-aware strategies can mitigate long-term bias amplification without severely sacrificing accuracy.

In spite of notable progress in understanding popularity bias, fairness, and feedback loops, several key gaps persist. The long-term, dynamic effects of popularity bias on different user groups and the role of beyond-accuracy metrics in promoting fairness remain understudied. Additionally, while prior works have introduced simulation frameworks for RSs, many do not simultaneously incorporate popularity bias, fairness, calibration, and multiple quality dimensions under realistic iterative conditions. To the best of our knowledge, this work is the first to present a comprehensive dynamic simulation environment that jointly evaluates these aspects as they evolve. By analyzing the iterative changes in recommendation outcomes, our approach addresses an essential gap in the literature and provides a new avenue for future investigation.

3 The proposed feedback loop framework

This section introduces the simulation framework developed to model feedback loops in RSs. The framework is designed to analyze the dynamic interactions between recommendation algorithms and user profiles across multiple iterations, focusing on the effects of popularity bias, fairness, and beyond-accuracy metrics on diverse user groups. The overall process is outlined in Algorithm 1. These serve as the basis for the explanation of each step in the framework.

```
 1: Initialize original user-item rating matrix $R$
 2: Classify items into Head ($H$) and Tail ($T$) using Pareto principle
 3: Segment users into Popular-, Diverse-, and Niche-focused groups ($G$)
 4: **for** $i = 1$ to the iteration count $t$ **do**
 5:     **for** each user $u$ **do**
 6:         Predict ratings $\hat{r}_{u,i}$ for all items using any CF algorithm
 7:         Generate top-$N$ recommendation list: $L_u$
 8:         Evaluate $L_u$:
              Compute the chosen metric for each user $u$
              Aggregate the metric values of users within each group $G$ defined in Step 3
 9:         **Simulate consumption:**
              Compute profile size: $k_u$
              Normalize all $k_u$ to $\tilde{k}_u$
              Compute # of items to fill: $c_u$
              Compute user-specific volatility $\sigma_u$ from $u$'s existing ratings
              Traverse $L_u$ from top to bottom and update up to $c_u$ items as follows:
                *If the item is unrated by u (new consumption):* set its rating using an item-based similarity–weighted
                estimate from $u$'s past ratings, add a small Gaussian noise scaled by $\sigma_u$, clip to the used rating
                scale, and write to $R'$
                *If the item is already rated by u (repeat consumption):* take the last observed rating of $u$ for that
                item, add a small Gaussian noise scaled by $\sigma_u$, clip to the used rating scale, and write to $R'$
10:         Update $R'$
11:     **end for**
12:     Update user-item matrix $R \leftarrow R'$
13: **end for**
```

Algorithm 1 Proposed Feedback Loop Framework.

The framework consists of the following steps, as outlined in Algorithm 1. In the following, we describe these steps in detail.

1. ***Initialize User-Item Rating Matrix***: The framework begins with the original user-item rating matrix *R*, where each entry $r_{u,i}$ represents the interaction between user *u* and item *i* (*line 1*). This matrix serves as the foundation for predicting user preferences and generating top-*N* recommendations.

2. ***Classify Items into Head and Tail Groups***: Items are first sorted in descending order based on their rating frequencies (f_i). Following the well-known Pareto principle (Sanders, 1987), a threshold is set to identify the top 20% of total ratings. For example, if the dataset contains a total of 1000 ratings, the first L items with cumulative frequencies $\sum f_i \leq 200$ are categorized as Head (H), while the remaining items are classified as Tail (T) (*line 2*). This method explicitly incorporates rating frequency into the classification process, ensuring a clear differentiation between popular and niche items.
3. ***Segment Users into Groups:*** Users are segmented into three groups: *Popular-focused*, *Diverse-focused*, and *Niche-focused*, based on the proportion of popular items (H) in their profiles (*line 3*). For each user, the ratio of interactions with popular items to their total interactions is computed using (1):

$$p_u = \frac{|I_u^H|}{|I_u|} \tag{1}$$

where $|I_u^H|$ is the number of popular items (*H*) in user u's profile, $|I_u|$ is the total number of items rated by user u, and p_u is the user's inclination toward popular items, ranging from 0 (no rating for popular items) to 1 (all ratings are with popular items). To classify users into groups, we assume that p_u values follow a normal distribution and apply the *Assumption of Normality*, as described in Yalcin and Bilge (2022a). This grouping strategy is defined as follows:

- *Popular-focused:* Users with $p_u > (\mu + \sigma)$, where μ is the mean and σ is the standard deviation of p_u values.
- *Niche-focused:* Users with $p_u < (\mu - \sigma)$, indicating a stronger preference for niche (*T*) items.
- *Diverse-focused:* Users with $(\mu - \sigma) \leq p_u \leq (\mu + \sigma)$, reflecting a balanced preference between popular and niche items.

This grouping strategy offers several advantages over straightforward approaches, such as assigning users to groups based on fixed thresholds (e.g., selecting the top 20% of p_u values) or using absolute criteria (e.g., classifying users with $p_u > 0.7$ as *Popular-focused*). While these methods are simple, they rely on arbitrary cutoffs that may not align with the empirical distribution of p_u in a given dataset and can obscure meaningful behavioral differences. In contrast, the *Assumption of Normality* provides data-adaptive boundaries grounded in the observed mean and variability of p_u. However, we note that this approach does not enforce equal group sizes and may yield skewed segments; this is an expected outcome that reflects the natural heterogeneity of user preferences rather than a methodological imbalance.

4. ***Iteration Steps***: In this step, the framework performs an iterative feedback loop that encompasses generating recommendations, evaluating their quality, and updating user profiles (*lines 6-10*). This process is designed to dynamically capture how recommendations evolve over successive iterations, enabling a comprehensive analysis of metrics such as fairness, accuracy, and beyond-accuracy dimensions across diverse user groups. Each iteration consists of multiple sub-steps, outlined below, which collectively model the real-world dynamics of RSs.

1. *Compute Predictions:* Any CF algorithm is applied to compute predicted ratings $\hat{r}_{u,i}$ for all user-item pairs in the dataset. These predicted values represent the system's estimation of how each user *u* would rate each item *i*, forming the foundation for generating personalized recommendation lists.
2. *Generate Top-N Recommendation Lists:* Using the predicted ratings, a ranked top-*N* list L_u is generated for each user *u*. This list represents the most relevant items for the user.
3. *Evaluate Recommendations:* The top-*N* recommendation lists are evaluated using any evaluation metrics (e.g., fairness, accuracy, and beyond-accuracy). These evaluations are conducted in two stages. First, the metrics are computed individually for each user *u* based on their recommendation list L_u. Second, for each predefined user group in Step 3, the group-level metrics are calculated by averaging the individual metrics of all users within that group. For each user-level metric M, individual user scores (M_u) are first computed, and group-level results (M_G) are then obtained by taking the average of all users in the corresponding group. This aggregation provides a representative value of the metric for each user segment without requiring additional weighting or normalization. On the other hand, for system-level metrics such as long-tail coverage explained in Section 4.4.3, each user group is treated as a single entity (i.e., a virtual system), and the metric is calculated over the aggregated recommendations for all users in the group. This approach allows system-level characteristics to be analyzed at the group level, providing insights into how such metrics differ across user groups.
4. *Simulate User Consumption:* This step is a cornerstone of our proposed framework, enabling the creation of a feedback loop mechanism that dynamically updates user profiles with synthetic ratings. By modeling user interaction with recommendations, this process iteratively evaluates how recommendations evolve over time. The simulation proceeds as follows:

    1. Compute the profile size $k_u = |P_u|$, where P_u represents the set of items rated by user *u*. This step determines the number of items in the user's current profile and provides a baseline for calculating the number of synthetic ratings to be added.
    2. Normalize k_u values into the range $[0, 1]$ using standard min–max normalization, where the smallest and largest profile sizes across users are mapped to 0 and 1, respectively. The resulting normalized value $\tilde{k}_u$ represents the scaled activity level of user u. This step ensures consistent scaling across users and prevents those with extremely large or small profiles from disproportionately influencing the generation of synthetic ratings in later stages. This normalization step provides a proportional scaling mechanism to map varying user profile sizes into a consistent range, ensuring numerical comparability when generating synthetic ratings. Users with larger profiles are thus assigned proportionally more synthetic ratings, reflecting their higher activity levels, while users with smaller profiles receive fewer. This approach maintains a stable ratio of feedback generation across users with different activity levels without distorting the underlying behavioral differences. Without normalization, large disparities

in profile sizes could lead to uneven weighting in synthetic rating generation, potentially affecting the analysis of recommendation performance and fairness.

3. Compute the number of new ratings to be generated for each user, c_u, as:

$$c_u = N \cdot \tilde{k}_u \tag{2}$$

where N is the size of the recommendation list, and c_u represents the number of synthetic ratings to be added to the user's profile. This calculation ensures that the synthetic ratings scale consistently with the normalized profile size, making the feedback loop realistic and equitable.

4. Add r_u as synthetic ratings to the top c_u items from the user's top-N recommendation list L_u. This step expands the user's profile by simulating interactions with the most highly recommended items while maintaining consistency with their historical rating behavior. The following step explains in detail how these synthetic ratings are generated.
5. For each user *u* and item *i*, rating generation is performed according to two distinct cases: *repeat consumption* and *new consumption*.

- ***Repeat consumption***: If user *u* has already rated item *i* in the past, the most recent rating $r_{u,i}^{last}$ is retrieved, and a small Gaussian noise term $\epsilon \sim \mathcal{N}(0, \sigma_u^2)$ is added:

$$r'_{u,i} = r_{u,i}^{last} + \epsilon \tag{3}$$

where σ_u is the user-specific rating volatility (standard deviation of ratings in *u*'s profile). This noise models contextual variations in user preference without artificially narrowing the rating distribution. The explicit modeling of repeat consumption in this way is consistent with prior work addressing re-consumption behavior in RSs (Chen et al., 2016).

- ***New consumption***: If user *u* has never rated item *i*, the rating is generated using an item-based CF (ICF) approach. Specifically, the cosine similarity between item *i* and all items in I_u (the set of items rated by *u*) is computed. The generated rating is then calculated as the weighted average of the ratings given by *u* to similar items. After prediction, a Gaussian noise term $\epsilon \sim \mathcal{N}(0, \sigma_u^2)$ is added to introduce realistic variability in the user's response:

$$\hat{r}_{u,i} = \frac{\sum_{j \in I_u} \text{sim}(i,j) \cdot r_{u,j}}{\sum_{j \in I_u} |\text{sim}(i,j)|} + \epsilon \tag{4}$$

In both *repeat* and *new consumption* cases, the resulting final value is clipped to the valid rating scale (e.g., [1, 5]) to avoid generating outlier ratings. This rating generation strategy offers two key advantages. First, it explicitly separates repeat and new consumption scenarios, yielding a more realistic simulation of user behavior. In repeat consumption, users' historical preferences are preserved while controlled Gaussian noise introduces natural variability. In new consumption, similarity-weighted predictions ensure consistency with established taste profiles, while added noise accounts for contextual deviations. A critical methodological choice is not to reuse the CF algorithm's predictions for synthetic rating generation. Ranking-oriented CF models often produce outputs that deviate from the dataset's rating scale and variance structure, which would distort the simulation. Moreover, directly using the same model risks reinforcing its own biases across iterations, artificially amplifying popularity skew rather than reflecting realistic user behavior. To prevent this, the framework employs an ICF-based approach that generates scale-consistent estimates aligned with empirical distributions, while keeping the simulation neutral and independent of any single algorithm. Finally, by limiting the proportion of ratings updated per iteration and clipping predictions to the valid range, the framework preserves the global rating distribution and prevents the unrealistic narrowing of ratings over time.

    6. Update the user-item rating matrix to R', incorporating the synthetic ratings generated in the previous step. The updated matrix R' serves as the input for subsequent iterations of the feedback loop, enabling dynamic evaluation of recommendation performance and fairness over time.

5. ***Matrix Update and Loop Re-initialization***: The updated rating matrix R' generated at the end of each iteration is fed back into Step 4, where all subsequent processes, such as prediction, recommendation generation, evaluation, and user profile updates, are repeated using this updated matrix. This iterative process continues for t iterations, dynamically updating user profiles and recommendations while tracking changes in metrics over time. By iteratively refining the rating matrix and analyzing the evolving recommendations, this framework enables a comprehensive analysis on RSs. It provides valuable insights into optimizing recommendation quality for diverse user groups in dynamic environments. To further clarify the workflow, Fig. 1 illustrates an overview

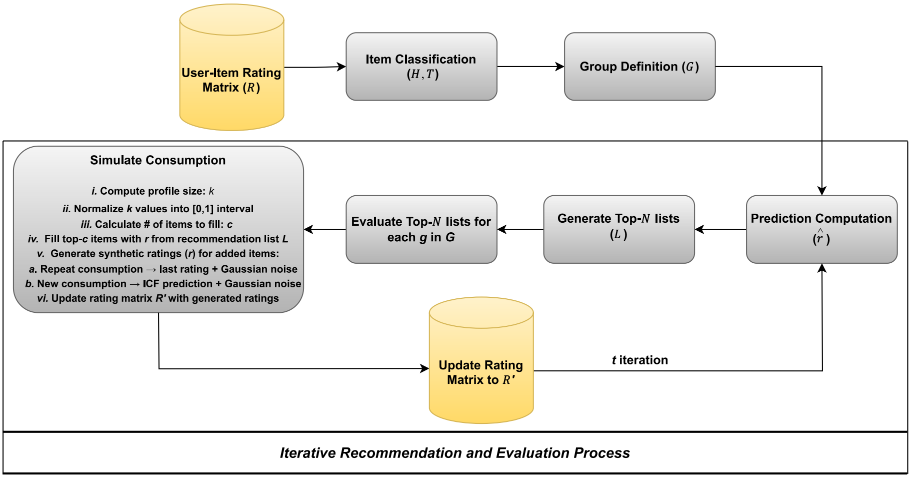
User-Item Rating Matrix (R)
Item Classification (H, T)
Group Definition (G)
Simulate Consumption
i. Compute profile size: k
ii. Normalize k values into [0,1] interval
iii. Calculate # of items to fill: c
iv. Fill top-c items with r from recommendation list L
v. Generate synthetic ratings (r) for added items:
a. Repeat consumption → last rating + Gaussian noise
b. New consumption → ICF prediction + Gaussian noise
vi. Update rating matrix R' with generated ratings
Evaluate Top-N lists for each g in G
Generate Top-N lists (L)
Prediction Computation ($\hat{r}$)
Update Rating Matrix to R'
t iteration
Iterative Recommendation and Evaluation Process

Fig. 1 Overview of the proposed framework for feedback loop

of the proposed feedback loop mechanism, offering a clear understanding of its operational steps.

4 Experimental setup

This section provides detailed information about the followed experimentation methodology, utilized datasets, CF algorithms and evaluation metrics, respectively.

4.1 Experimentation methodology

The recommendation lists were generated using a *user-level leave-one-out (LOO) cross-validation* protocol (Yalcin and Bilge, 2022a). In each evaluation step, one user is designated as the test user, while all remaining users constitute the training set. The recommendation algorithms are trained solely on the training users' interactions, after which all candidate items for the test user are scored and ranked. The top-N items with the highest prediction scores ($N = 10$ in this study) are then selected to form that user's recommendation list. This procedure is repeated for every user in the dataset, ensuring complete coverage and preventing any information leakage from a user's interactions into the model that is evaluated on that same user.

Following this methodology, experiments were conducted using the proposed framework described in Section 3, iterating over $t = 10$ iterations. During these iterations, user profiles were dynamically updated based on their interaction with recommendations, simulating a feedback loop mechanism. The values for each iteration were calculated for the *diverse-*, *popular-*, and *niche-focused* user groups using the metrics outlined in Section 4.4. This iterative approach allows us to comprehensively analyze how recommendation quality evolves over time for different user groups. By leveraging this methodology, we aim to capture the dynamic effects of popularity bias and evaluate the performance of algorithms in adapting to diverse user preferences.

To ensure reproducibility, all experiments were implemented under a deterministic LOO setup, in which the same users are held out for testing in every iteration, resulting in identical train–test splits across runs. This design eliminates sampling variance inherent in random-split methods, meaning that re-running the experiment with identical configurations yields the same outcomes. To confirm numerical stability, each algorithm was executed three times with different random seeds, and the observed variations were negligible. Therefore, only one representative set of results is reported.

4.2 Datasets

In this study, two datasets, MovieLens-1M (MLM) and Personality 2018 (Per), were used to examine popularity bias, unfairness, and calibration in RSs. The MLM contains 1,000,209 ratings provided by 6,040 users for 3,900 movies (Harper and Konstan, 2015), along with demographic details (e.g., age, gender, occupation) and movie metadata (e.g., genres, release years). The Per dataset, on the other hand, consists of 911,369 ratings from 1,780 users for 7,228 movies on a 10-star scale (Nguyen et al., 2018). It captures "Big Five" personality traits, i.e., openness, conscientiousness, extraversion, agreeableness, and neuroticism, thereby allowing an exploration of how individual traits influence fairness and calibration. Note that because the original dataset exhibits significant sparsity (approximately 98.4%), we focus on a subset in which every user and item has at least 20 ratings. Detailed user and item statistics, rating distributions, and sparsity rates for both datasets are provided in Table 1.

4.3 The utilized CF algorithms

In this study, we use a range of recently proposed CF algorithms employing different mechanisms, including probability-based approaches, and deep neural networks, to generate personalized recommendations. Specifically, we select Weighted Bayesian Personalized Ranking (WBPR) (Gantner et al., 2012), Neural Matrix Factorization (NEUMF) (He et al., 2017), and the Variational Autoencoder for Collaborative Filtering (VAECF) (Liang et al., 2018) as our core recommendation models.

More specifically, WBPR extends the Bayesian Personalized Ranking (BPR) framework by introducing item-dependent weighting schemes to the ranking process. Unlike traditional rating-based models, BPR focuses on optimizing the relative order of items for each user, promoting items they are more likely to prefer. WBPR refines this approach by assigning weights to items, often based on their popularity or other contextual factors, allowing the model to control the prominence of popular versus niche items. NEUMF, on the other hand, combines the strengths of Generalized Matrix Factorization and Multi-Layer Perceptrons within a unified neural framework. This hybrid approach allows NEUMF to model both linear interactions between users and items and complex non-linear patterns through deep learning. By integrating these two techniques, NEUMF can capture a wide range of user preference signals, including subtle or implicit cues that traditional methods might miss. Finally, VAECF utilizes deep generative models to represent user preferences in a probabilistic latent space. By encoding user-item interactions into a compressed representation and decoding them to reconstruct item ratings or implicit feedback, VAECF captures intricate patterns that traditional linear models struggle to identify. This non-linear structure allows the model to uncover nuanced relationships between users and items, enabling it to address complex preference dynamics. To ensure the reproducibility of our experiments, we implemented all algorithms using the *Cornac* (Salah et al., 2020) Python framework, which provides a robust environment for developing and evaluating RSs.

Hyperparameter Tuning Process *To ensure a fair and reproducible comparison across the evaluated CF algorithms, we performed systematic hyperparameter optimization using a grid search strategy. For each algorithm, the search was conducted within a predefined parameter grid derived from the ranges reported in the respective original papers and common practices in the RSs literature. During this process, we employed the nDCG as the*

Table 1 Detailed information about datasets

Dataset	$\|U\|$	$\|I\|$	$\|R\|$	Sparsity (%)	Avg_R	$\|R\|/\|I\|$	$\|R\|/\|U\|$
MLM	6,040	3,952	1,000,209	95.8	3.58	253.1	165.6
Per	1,780	7,228	911,369	92.9	3.41	126.09	512.01

Here, U is the set of users, I is the set of items, R is the set of all provided ratings, Avg_R is the average value of all ratings, and sparsity is defined as the ratio of unobserved ratings, i.e., $|U| \times |I| - |R|$, to the number of possible all ratings $|U| \times |I|$

objective metric and selected the configuration that yielded the highest validation nDCG score in each iteration of the simulation. This procedure was repeated for all datasets and models independently.

- For WBPR, we tuned the embedding dimension ($k \in \{64, 128\}$), the number of optimization iterations ($max\_iter \in \{100, 200\}$), the learning rate ($\eta \in \{1e^{-3}, 3e^{-4}\}$), and the regularization coefficient ($\lambda \in \{1e^{-4}, 1e^{-3}\}$).
- The NEUMF model was optimized by exploring the embedding size of the matrix factorization component ($num\_factors \in \{64, 128\}$), the multilayer perceptron structure ($layers \in \{[128, 64]\}$), the learning rate ($lr \in \{1e^{-4}, 1e^{-3}\}$), and the batch size ($batch\_size = 1024$). The model was trained using the Adam optimizer with $num\_neg = 4$ negative samples and ReLU activation functions.
- For VAECF, we examined the latent dimensionality ($k \in \{64, 128\}$), the number of training epochs ($n\_epochs \in \{20, 40\}$), the learning rate ($\eta \in \{1e^{-4}, 1e^{-3}, 1e^{-2}\}$), and the Kullback–Leibler divergence weight ($\beta \in \{0.5, 1.0\}$). The autoencoder structure was set to a single hidden layer with 200 units, using ReLU activation and multinomial likelihood.

All algorithms were implemented in the *Cornac* framework and trained with fixed random seeds for reproducibility. Each grid configuration was evaluated independently, and the parameter combination with the best *n*DCG performance was retained for the final analysis. Importantly, the tuning process was designed to achieve model-specific optimal performance within the computational limits of our experimental setup. While the explored parameter ranges are consistent with the literature, we acknowledge that different search spaces or tuning techniques might further improve the performance of certain algorithms.

4.4 Evaluation metrics

In this study, we evaluate how users are differently affected by the popularity bias of recommenders within the proposed feedback loop framework, utilizing a diverse set of metrics grouped into three main categories: popularity-bias and calibration (Boratto et al., 2022), accuracy (McNee et al., 2006), and beyond-accuracy metrics (Yalcin and Bilge, 2022a). Each category highlights a distinct aspect of the system's performance, enabling a comprehensive evaluation that incorporates both traditional measures and advanced perspectives.

4.4.1 Calibration and popularity-bias metrics

Fairness metrics examine whether a RS provides balanced outcomes for both users and items. From the user side, they assess alignment with historical preferences regarding item popularity; from the item side, they evaluate whether exposure is equitably distributed across popular and niche items. In this study, we employ two complementary metrics that capture group- and item-level fairness dimensions.

- **Group Average Popularity (ΔGAP)**: ΔGAP (Abdollahpouri et al., 2019b) quantifies how recommendations shift the average popularity of items for different user groups. It compares the mean popularity in users' historical profiles ($GAP_p(g)$) and in the corresponding recommendation lists ($GAP_r(g)$):

$$GAP_p(g) = \frac{\sum_{u \in g} \sum_{i \in p_u} \phi(i)}{|p_u| \cdot |g|}, \quad GAP_r(g) = \frac{\sum_{u \in g} \sum_{i \in r_u} \phi(i)}{|r_u| \cdot |g|} \tag{5}$$

where $\phi(i)$ is the relative popularity of item i. The final difference, $\Delta GAP = (GAP_r(g) - GAP_p(g))/GAP_p(g)$, indicates the extent of popularity shift. Values close to zero reflect fairer alignment between the algorithm's output and users' historical preferences, while positive (negative) values indicate a bias toward more (less) popular items.

- **Average Popularity of Recommended Items (APRI):** The *APRI* metric (Yalcin, 2022) measures the average popularity of items in the top-N list, focusing on item-exposure fairness:

$$APRI = \frac{1}{|N|} \sum_{i \in N} P_i \tag{6}$$

where P_i denotes the fraction of users who interacted with item i. A lower *APRI* value signifies that the RS favors less popular (more diverse) items, indicating reduced popularity bias. Importantly, *APRI* is conceptually related to the *Novelty* measure (Lü et al., 2012; Wang et al., 2016), which captures exposure to less frequent items:

$$novelty = \frac{1}{mN} \sum_{u=1}^{m} \sum_{i \in L_u} d_i \tag{7}$$

where L_u is the top-N list of user u, m the total number of users, and d_i the degree of item i (i.e., the number of users that rated the item *i*). Similar to this formulation, *APRI* reflects the mean popularity of recommended items and can thus be interpreted as an inverse indicator of novelty and a direct measure of popularity bias.

4.4.2 Accuracy metric

Accuracy metrics evaluate how effectively a RS predicts user preferences. In this study, we employ the *Normalized Discounted Cumulative Gain (nDCG)* as the primary accuracy measure, a widely used metric for assessing ranked recommendation quality (Chia et al., 2022).

For a user u and their top-N recommendation list $\{i_1, i_2, \ldots, i_N\}$, the Discounted Cumulative Gain (DCG) is defined as:

$$DCG_u^N = r_{u,i_1} + \sum_{n=2}^{N} \frac{r_{u,i_n}}{\log_2(n)} \tag{8}$$

where $r_{u,i}$ denotes the relevance or rating of item i for user u. The ideal DCG ($IDCG_u^N$) represents the maximum possible gain when items are perfectly ranked by relevance. The normalized form is then computed as:

$$nDCG_u^N = \frac{DCG_u^N}{IDCG_u^N} \tag{9}$$

Averaging $nDCG_u^N$ across users yields the final system-level score. Higher values indicate better ranking consistency with users' true preferences.

4.4.3 Beyond-accuracy metrics

Beyond-accuracy metrics extend evaluation beyond traditional accuracy measures by considering aspects such as diversity, fairness, and exposure balance (Castells et al., 2022; Vargas and Castells, 2011). These metrics ensure that RSs not only produce accurate predictions but also deliver varied and equitable experiences for different users (Abdollahpouri et al., 2019b).

- **Average Percentage of Long-Tail Items (APLT):** APLT (Abdollahpour, 2020) measures the proportion of items in a user's recommendation list that belong to the long-tail segment of the catalog. Based on the Pareto principle (Sanders, 1987), items are divided into "head" (top 20%) and "tail" (remaining 80%) categories. The metric is defined as:

$$APLT_u = \frac{|\{i \,|\, i \in (N_u \cap T)\}|}{|N_u|} \tag{10}$$

where N_u is the set of items recommended to user u, and T is the set of tail items. A higher APLT score indicates stronger inclusion of less popular items, promoting diversity and mitigating popularity bias.

- **Entropy:** Entropy (Elahi et al., 2021c) quantifies how evenly the recommendation opportunities are distributed across items. It measures the diversity of item exposure within all top-N lists combined:

$$Entropy = -\sum_{i \in K} \Pr(i) \log_2 \Pr(i) \tag{11}$$

where $\Pr(i)$ is the relative frequency of item i appearing in the aggregated recommendation lists. A higher Entropy value reflects more balanced exposure, avoiding over-representation of a few popular items.

- **Long-Tail Coverage (LTC):** LTC (Abdollahpour, 2020) evaluates how comprehensively a system covers the long-tail portion of the catalog. Let $\mathbb{N}$ denote the union of all top-N lists (duplicates removed) and T the set of tail items. LTC is defined as:

$$LTC = \frac{|I_{\mathbb{N} \cap T}|}{|T|} \tag{12}$$

where $|T|$ is the total number of tail items. A higher LTC score indicates broader inclusion of niche content, enhancing fairness and catalog utilization.

5 Experiment results

This section presents a comprehensive evaluation of the proposed framework by analyzing the performance of three CF algorithms, NEUMF, VAECF, and WBPR, across two datasets, MLM and PER, over $t = 10$ iterations. The results are evaluated using the fairness, accuracy, and beyond-accuracy metrics described in Section 4.4, providing a multidimensional perspective on recommendation outcomes. For each iteration, the concrete results are analyzed, along with the proportional changes observed between the 1st and 10th iterations. The primary goal of these analyses is to understand how the three user groups, *popular-*, *diverse-*, and *niche-focused*, are impacted differently by the recommendation framework over multiple iterations, with a specific focus on how the iterative feedback loop influences the metrics across user groups. This approach allows for an in-depth understanding of the dynamics introduced by the simulation and the varying effects on different user profiles.

In the following, we first describe the characteristics of the user groups constructed using the strategy outlined in Section 3. Subsequently, we provide an analysis showing how the stability of the datasets are changed across simulation iterations. Finally, we present and discuss the experimental results obtained for these groups across fairness, accuracy, and beyond-accuracy metrics.

5.1 Characteristics of the constructed user groups

Before discussing the differences in recommendation quality among user groups within the context of the proposed framework, this section introduces the characteristics of the constructed groups. The user groups are constructed using the 3rd step of the proposed framework in Section 3. An overview of the key characteristics of these groups, including the number of users, their ratio of interactions with head items, and their profile size, is presented in Table 2. These characteristics reveal notable differences among the groups, shedding light on their unique behaviors and preferences.

Popular-focused group, demonstrates the highest interaction with head items, with a ratio of 93% in MLM and 98% in PER. These users exhibit preferences strongly aligned with popular content, but their profile sizes remain relatively small, indicating less frequent

Table 2 Characteristics of the constructed groups

Dataset	Group	# Users	Ratio of Head Items (%)	Profile Size
MLM	*Popular-focused*	953	93	49
	Diverse-focused	4,223	84	148
	Niche-focused	864	70	380
PER	*Popular-focused*	288	98	71
	Diverse-focused	1,286	86	444
	Niche-focused	206	68	1,554

interactions with the system compared to other groups. *Diverse-focused* group, on the other hand, strikes a balance between head and tail items, with a lower ratio of head items and moderate profile sizes. This group reflects users who interact with a mix of popular and less popular content, making them valuable for algorithms aiming to cater to diverse preferences.

Niche-focused group, stands out with the largest profile sizes across both datasets, as can be followed by Table 2. These users interact predominantly with niche items, as evidenced by their low ratio of head items. Their high engagement levels highlight their importance for the system as the most active users. These users represent the core audience of such platforms, as their frequent interactions can significantly influence system metrics. However, this also underscores the critical need to prioritize their satisfaction, as they are the most likely to notice and be impacted by algorithmic shortcomings.

5.2 Stability of dataset characteristics across simulation iterations

To establish methodological rigor, we systematically analyzed the descriptive statistics of the datasets at the initial (i.e., baseline) and after the 10th iteration (see Table 3). The analysis focused on measures of central tendency (mean, μ), variability (variance, σ^2), and dispersion (standard deviation, σ), in order to assess the stability of the dataset structure throughout the process. As shown in Table 3, the descriptive statistics before and after the simulation remain highly consistent. While minor fluctuations in μ, σ^2, and σ were observed, paired *t*-tests confirmed that these differences were statistically non-significant ($p > 0.05$). This indicates that the overall structure and distributional characteristics of the datasets were preserved, thereby reinforcing the robustness and reliability of the simulation outcomes. Note that the results presented in Table 3 correspond to the VAECF algorithm. However, the same overall trends are observed across the other algorithms as well. For the sake of clarity, we report only the VAECF results here.

As reported in Table 3, descriptive measures exhibited only marginal fluctuations across groups, with means (μ), variances (σ^2), and standard deviations (σ) preserved over time. Importantly, these findings demonstrate that the simulation framework successfully established a controlled environment while preserving the intrinsic statistical properties of the datasets. The *popular-focused* group retained relatively higher means, whereas the *diverse-*

Table 3 Descriptive statistics (mean μ, variance σ^2, and standard deviation σ) of overall and group-based ratings across the baseline and 10th iterations, reported separately for the MLM and Per datasets

	MLM		Per	
Iteration	*Baseline*	*10th*	*Baseline*	*10th*
μ_{global}	3.581	3.571	3.406	3.403
$\mu_{\mathrm{popular-focused}}$	3.911	3.880	3.966	3.954
$\mu_{\mathrm{diverse-focused}}$	3.711	3.693	3.583	3.579
$\mu_{\mathrm{niche-focused}}$	3.430	3.421	3.277	3.276
$\sigma^2_{\mathrm{global}}$	1.062	1.097	0.811	0.810
$\sigma^2_{\mathrm{popular-focused}}$	0.981	1.013	0.708	0.722
$\sigma^2_{\mathrm{diverse-focused}}$	1.062	1.085	0.825	0.830
$\sigma^2_{\mathrm{niche-focused}}$	1.150	1.172	0.872	0.879
σ_{global}	1.010	1.022	0.867	0.863
$\sigma_{\mathrm{popular-focused}}$	0.969	0.983	0.795	0.803
$\sigma_{\mathrm{diverse-focused}}$	1.011	1.021	0.876	0.879
$\sigma_{\mathrm{niche-focused}}$	1.052	1.060	0.905	0.908

and *niche-focused* groups reflected their expected rating distributions without substantial deviations. This consistency highlights that the simulation design did not distort or bias the original data structure, thereby providing a valid and methodologically rigorous foundation for further analysis. The stability observed across both aggregate- and group-level structures confirms the robustness and reliability of the simulation process, ensuring that the reported outcomes are dependable and reproducible.

5.3 Experiment results

Tables 4 and 5 present a comprehensive summary of all experimental outcomes obtained from the two benchmark MLM and PER datasets. Each table reports the baseline and the 10th iteration results for the utilized three CF algorithms, i.e., VAECF, WBPR, and NEUMF, across the user groups identified in the simulation: G_1 (*popular-focused*), G_2 (*diverse-focused*), and G_3 (*niche-focused*). The metrics encompass multiple evaluation dimensions,

Table 4 All obtained results for baseline, 10th iteration, and change ratio (%) across different user groups on the MLM dataset

		VAECF			WBPR			NEUMF		
		G_1	G_2	G_3	G_1	G_2	G_3	G_1	G_2	G_3
ΔGAP	*Baseline*	61.233*	57.413	34.537	77.009*	94.664	107.477	88.620*	132.603	155.157
	10th	64.484*	62.499	44.081	77.698*	99.635	115.399	92.198*	142.860	174.460
	Change Ratio (%)	5.309	8.860	27.635	0.895	5.252	7.371	4.037	7.736	12.441
APRI	*Baseline*	0.355*	0.244	0.139	0.389*	0.300	0.211	0.413*	0.356	0.262
	10th	0.381*	0.270	0.162	0.400*	0.314	0.224	0.441*	0.385	0.294
	Change Ratio (%)	7.428	10.415	16.462	2.948	4.714	5.831	6.611	8.169	12.019
*n*DCG	*Baseline*	0.637	0.636	0.624	0.610	0.627	0.617	0.556*	0.610	0.630
	10th	0.680*	0.657	0.622	0.625*	0.617	0.601	0.577*	0.608	0.606
	Change Ratio (%)	6.748	3.323	-0.367	2.517	-1.648	-2.622	3.656	-0.281	-3.825
APLT	*Baseline*	0.108*	0.458	0.814	0.031*	0.271	0.592	0.010*	0.125	0.439
	10th	0.088*	0.386	0.761	0.030*	0.262	0.574	0.007*	0.102	0.401
	Change Ratio (%)	-18.098	-15.683	-6.532	-4.282	-3.219	-3.031	-31.753	-18.408	-8.604
Entropy	*Baseline*	0.549*	0.720	0.812	0.487	0.622	0.698	0.443*	0.545	0.654
	10th	0.534*	0.710	0.778	0.485	0.612	0.667	0.443*	0.533	0.633
	Change Ratio (%)	-2.728	-1.302	-4.142	-0.400	-1.638	-4.394	0.059	-2.237	-3.117
LTC	*Baseline*	0.0550*	0.2805	0.3282	0.0143*	0.1196	0.1271	0.0065*	0.0737	0.1013
	10th	0.0648*	0.3039	0.3357	0.0135*	0.1130	0.1103	0.0064*	0.0698	0.0928
	Change Ratio (%)	17.979	8.328	2.275	-5.820	-5.471	-13.256	-0.026	-5.325	-8.451

G_1: Popular-focused, G_2: Diverse-focused, G_3: Niche-focused * Significant at the 95% confidence level between G_1 and G_3, tested separately for *baseline* and *10th iteration*

Table 5 All obtained results for baseline and 10th iteration, and change ratio (%) across different user groups on the PER dataset

		VAECF			WBPR			NEUMF		
		G_1	G_2	G_3	G_1	G_2	G_3	G_1	G_2	G_3
ΔGAP	*Baseline*	74.077*	204.163	296.065	70.025*	148.923	222.064	71.775*	175.626	289.346
	10th	76.463*	223.662	339.558	72.423*	155.232	233.411	73.735*	189.379	317.941
	Change Ratio (%)	3.221	9.551	14.690	3.423	4.237	5.110	2.731	7.831	9.882
APRI	*Baseline*	0.758*	0.745	0.644	0.742*	0.621	0.465	0.749*	0.680	0.563
	10th	0.783*	0.789	0.705	0.755*	0.632	0.482	0.795*	0.728	0.609
	Change Ratio (%)	3.279	5.936	9.349	1.714	1.812	3.515	6.180	7.045	8.159
*n*DCG	*Baseline*	0.634	0.676	0.640	0.657*	0.743	0.764	0.624*	0.720	0.761
	10th	0.665*	0.670	0.657	0.696*	0.744	0.725	0.640*	0.699	0.704
	Change Ratio (%)	4.872	-0.869	2.593	5.942	0.069	-5.076	2.646	-2.915	-7.462
APLT	*Baseline*	0.000*	0.000	0.526	0.000*	0.066	0.308	0.000*	0.016	0.143
	10th	0.000*	0.022	0.437	0.000*	0.065	0.307	0.000*	0.012	0.127
	Change Ratio (%)	–	–	-16.913	–	-2.772	-0.094	–	-24.482	-10.795
Entropy	*Baseline*	0.271*	0.322	0.411	0.328*	0.499	0.593	0.312*	0.426	0.516
	10th	0.264*	0.292	0.378	0.318*	0.489	0.593	0.289*	0.402	0.481
	Change Ratio (%)	-2.500	-9.291	-7.953	-3.243	-2.005	0.000	-7.341	-5.744	-6.664
LTC	*Baseline*	0.0000*	0.0000	0.0026	0.0000*	0.0253	0.0332	0.0000*	0.0078	0.0171
	10th	0.0000*	0.0077	0.0134	0.0000*	0.0243	0.0312	0.0000*	0.0071	0.0143
	Change Ratio (%)	–	–	422.073	–	-3.963	-6.016	–	-9.485	-16.354

G_1: Popular-focused, G_2: Diverse-focused, G_3: Niche-focused * Significant at the 95% confidence level between G_1 and G_3, tested separately for *baseline* and *10th iteration*

including fairness (ΔGAP and APRI), accuracy (*n*DCG), and beyond-accuracy aspects (APLT, Entropy, and LTC). For each metric, the percentage change between the initial and 10th iterations is also shown, providing a clear view of how algorithmic feedback loops influenced the models over time. In these tables, asterisks (*) indicate statistically significant differences between G_1 and G_3 at the 95% confidence level. Statistical significance tests were conducted separately for both the *baseline* and the *10th iteration* results, as G_1 and G_3 represent the most distinct (extreme) user profiles in terms of popularity preference, while G_2 serves as an intermediate reference group. In the following, we discuss the key findings for each metric in detail using an itemized format.

- *Findings related to popularity-bias and calibration results*: The baseline ΔGAP values reveal systematic differences in exposure to popular items across user groups with distinct preference orientations. As shown in Tables 4 and 5, both datasets exhibit a consistent ordering (except for MLM–VAECF): G_1 users show the lowest ΔGAP, followed by G_2, while G_3 users present the highest values. This pattern demonstrates that popularity preferences directly shape fairness outcomes, with niche-oriented users being the most affected by algorithmic bias. Statistical significance tests confirm that the differences between G_1 and G_3 are significant at the 95% confidence level in both datasets, reinforcing that popularity-oriented bias produces measurable disparities across the most distinct user groups. Across algorithms, the magnitude of these disparities varies. In the MLM dataset, VAECF yields moderate inter-group differences but shows the largest relative increases by the 10th iteration, indicating that even users with balanced preferences are gradually pushed toward popular content. WBPR remains comparatively stable, whereas NEUMF exhibits stronger divergence, with ΔGAP values exceeding 150 for niche users, suggesting that deep neural architectures reinforce mainstream dominance. In the PER dataset, where popular items are more concentrated, these disparities become even more pronounced. By the 10th iteration, all models exhibit an overall increase in ΔGAP, with the most pronounced rises observed in G_3 users. This indicates that feedback loops disproportionately intensify inequalities for users already underrepresented in baseline recommendations. VAECF and NEUMF show the highest change ratios (above 10% and up to 25%), confirming that even complex models are not immune to feedback amplification of popularity bias. Overall, ΔGAP serves as a calibration-oriented metric, revealing how the system drifts away from users' original popularity alignment over time. Complementing these results, APRI captures the average popularity level of recommended items, offering an item-level perspective on popularity bias. Baseline APRI values follow a parallel ordering ($G_1 > G_2 > G_3$), showing that recommendations already favor popular content. VAECF and NEUMF produce the highest APRI scores in both datasets, while WBPR remains more balanced but still exhibits a skew toward popular items. In the PER dataset, where popular items dominate, APRI scores rise markedly for all groups, even for G_3 users who typically prefer niche content. At the 10th iteration, APRI values further increase across all groups, again most sharply for G_3. This upward trend indicates that feedback loops not only widen calibration gaps (as seen in ΔGAP) but also erode *novelty* (see (7)) by intensifying exposure to mainstream items. VAECF and NEUMF display the steepest increases, while WBPR maintains relative stability. The results collectively show that repeated recommendation cycles lead to a homogenization effect, reducing content diversity and users' opportunity for discovery. Taken together, ΔGAP and APRI expose complementary facets of popularity bias. ΔGAP reflects group-level calibration drift and fairness disparities, whereas APRI highlights the system's growing reliance on popular items and the consequent loss of novelty. Both metrics confirm that popularity bias is a structural property of the algorithms, reinforced through feedback, ultimately favoring G_1 users while disadvantaging G_3 and constraining the overall diversity of the recommendation space. To summarize, we outline the key insights gained from the experiments conducted using fairness-oriented metrics as follows.

Key findings for popularity-bias and calibration results:

- *Popular-focused Group (G_1):* Maintains strong and stable calibration from baseline to the 10th iteration, remaining least affected by popularity bias. Yet, persistently high APRI values indicate that their recommendations are dominated by popular items, offering limited novelty.
- *Diverse-focused Group (G_2):* Starts with balanced calibration but gradually drifts toward popular content, showing growing sensitivity to popularity bias. The concurrent increase in ΔGAP and APRI reflects declining diversity and partial loss of calibration over iterations.
- *Niche-focused Group (G_3):* Emerges as the most disadvantaged segment. Initially the least calibrated, this group also experiences the sharpest deterioration over time, with both ΔGAP and APRI steadily rising. These patterns indicate strong amplification of popularity bias and a marked erosion of calibration, diversity, and novelty.

- *Findings related to accuracy results*: At the baseline stage, ranking accuracy appears relatively balanced across groups in several cases, yet G_3 users exhibit higher nDCG values in many settings. In particular, their advantage is statistically significant in specific scenarios, notably with WBPR and NEUMF models on the PER dataset, and NEUMF on the MLM dataset. This indicates that, despite their limited exposure to popular items, niche users can initially achieve higher local ranking precision, likely due to concentrated preference profiles. As can be followed by Tables 4 and 5, the general tendency shifts after ten iterations. Except for VAECF, where G_1 users achieve the highest relative improvement, G_3 shows a noticeable decline in ranking accuracy across most models, while G_2 remains relatively stable. Nevertheless, G_3 retains comparatively high absolute nDCG values in certain configurations, indicating that niche-oriented users can still achieve strong ranking precision despite feedback-induced degradation. When viewed proportionally, however, G_3 becomes the most disadvantaged group: its relative accuracy changes are consistently lower (and often negative), whereas G_1 shows the strongest growth across iterations. This pattern suggests that iterative feedback increasingly favors users aligned with popular content, amplifying disparities in model calibration and ranking outcomes. The results conclude that accuracy improvements are unevenly distributed, with G_3 losing momentum despite maintaining relatively high absolute accuracy in some cases. To summarize, we outline the key insights gained from the experiments conducted using accuracy metric as follows.

Key findings for accuracy results:

- *Popular-focused Group (G_1):* Achieves the highest relative accuracy gains across iterations. Feedback loops reinforce their alignment with popular content, consistently improving ranking quality.
- *Diverse-focused Group (G_2):* Maintains moderate and stable accuracy. While some improvement occurs, feedback effects are weaker, and ranking benefits remain limited compared to G_1.
- *Niche-focused Group (G_3):* Initially attains high or even superior absolute accuracy in several scenarios, yet experiences the steepest proportional declines over iterations. Despite retaining strong baseline precision, feedback loops progressively disadvantage this group under popularity bias.

- *Findings related to beyond-accuracy results:* The APLT metric captures how extensively algorithms expose users to long-tail content. As presented in Tables 4 and 5, baseline values show a clear stratification across user groups: G_3 users receive the highest long-tail exposure, G_2 users moderate exposure, and G_1 users are almost entirely limited to head items. In fact, for the PER dataset, all algorithms yield an APLT value of zero for G_1, indicating a complete absence of long-tail content in their recommendation lists. These baseline patterns confirm that recommendation outputs strongly mirror users' popularity orientations rather than compensating for them. Another contributing factor to these low or zero values, particularly in the PER dataset, is the smaller size of the tail-item subset compared to MLM, which inherently limits opportunities for long-tail exposure. After ten iterations, a consistent contraction in long-tail exposure emerges across all groups and models. In the MLM dataset, APLT decreases universally, with the most severe reductions observed for G_1, indicating a complete reinforcement of head-item dominance. G_2 and G_3 also experience measurable declines, though less steep, implying that iterative feedback narrows content variety even for users initially inclined toward niche items. In the PER dataset, however, G_1 users continue to exhibit zero APLT values, confirming a total absence of long-tail recommendations both at baseline and after iteration. This stability at zero underscores that once the system fully converges to head-item dominance, no recovery of long-tail exposure occurs, further reinforcing popularity bias over time. Overall, APLT trends reveal that iterative feedback systematically suppresses long-tail coverage for all user groups, driving recommendation lists toward a popularity-dominated equilibrium. While the steepest relative contractions occur for G_1 in MLM, reflecting a complete reinforcement of head-item dominance, G_3 users lose much of their initial advantage in long-tail diversity as their exposure narrows across iterations. In the PER dataset, the persistence of zero APLT values for G_1 further illustrates that once long-tail representation collapses, the system fails to recover it, underscoring how feedback dynamics entrench popularity bias and erode diversity across all preference groups. Entropy, which captures the internal diversity of recommendation lists, reveals how evenly items are distributed across the catalog. Across both datasets, baseline results follow a consistent ordering: G_3 users demonstrate the highest entropy, G_2 users occupy an intermediate position, and G_1

users remain lowest. This structure indicates that niche-oriented users initially benefit from broader exposure to unique items, while popular users receive more concentrated recommendations dominated by head content. Following iterative feedback, entropy decreases for nearly all groups and algorithms, showing that recommendation diversity declines over time as systems increasingly reinforce popular content. In the MLM dataset, all groups exhibit mild-to-moderate declines, strongest for G_3 and G_2, whereas G_1 experiences relatively stable diversity. In the PER dataset, however, the pattern varies by algorithm: VAECF produces the sharpest drops, particularly for G_2, while WBPR and NEUMF maintain relatively smaller declines, suggesting that model structure can modulate, but not prevent, diversity erosion. Despite these nuances, the overall pattern is consistent: entropy reductions are systemic rather than incidental. In general, G_3 and G_2 users, while still showing the highest absolute diversity, lose a notable portion of their initial advantage. Meanwhile, G_1 users remain the least affected, reflecting the algorithmic tendency to preserve head-item concentration for popularity-oriented profiles. When considered alongside APLT, entropy trends confirm that reduced exposure to tail items translates into overall catalog compression, indicating that both local (list-level) and global (system-level) diversity deteriorate in parallel. The LTC metric evaluates the overall share of long-tail items represented in recommendation outputs, complementing APLT by capturing system-level inclusion rather than individual exposure. As shown in Tables 4 and 5, baseline LTC values align with APLT outcomes: in the MLM dataset, measurable coverage exists across all groups, whereas in the PER dataset, LTC remains near zero, especially for G_1 users, since these users receive no tail items in APLT and the tail-item subset itself is considerably smaller. This pattern indicates that dataset composition strongly constrains system-wide long-tail representation. After ten iterations, divergent trends emerge across algorithms. In the MLM dataset, LTC slightly increases under VAECF, suggesting a limited yet meaningful ability of this model to preserve or even expand long-tail representation. Conversely, both WBPR and NEUMF exhibit consistent declines, demonstrating that their learning processes progressively reinforce head-item dominance. Across user groups, G_3 users, who initially contribute most to long-tail representation, show the sharpest declines under WBPR and NEUMF, while VAECF partially mitigates this effect. G_1 and G_2 groups also experience moderate decreases, reflecting a gradual narrowing of catalog diversity over time. In the PER dataset, LTC values remain minimal across all iterations, with G_1 fixed at zero for every algorithm. For WBPR and NEUMF, both G_2 and G_3 demonstrate additional decreases (more pronounced for G_3), confirming that once long-tail inclusion is suppressed, the system fails to recover it. Overall, LTC trends indicate that iterative feedback amplifies head-item concentration in nearly all settings, diminishing long-tail representation at both user and system levels. Combined with APLT and entropy findings, the results demonstrate a coherent dynamic: reduced user-level exposure (APLT) leads to shrinking global tail coverage (LTC), while internal diversity (entropy) collapses in tandem. Together, these patterns reveal that feedback loops create a reinforcing cycle of popularity bias–where both personalization and diversity deteriorate as systems progressively converge toward head-item–dominated equilibria, particularly in datasets with limited tail diversity such as PER. To summarize, we outline the key insights gained from the experiments conducted using beyond-accuracy metrics as follows:

Key findings for beyond-accuracy results:

- *Popular-focused Group (G_1): Shows the weakest exposure to long-tail content, reflected by near-zero APLT and LTC values, particularly in the PER dataset, where no tail items appear in recommendations. Entropy remains low but relatively stable, indicating limited yet consistent diversity focused on popular items. Feedback loops further reinforce head-item dominance, keeping this group least affected by diversity loss but most restricted in content variety.*
- *Diverse-focused Group (G_2): Begins with moderate levels of long-tail exposure and internal diversity but undergoes steady declines across all metrics (APLT, Entropy, and LTC). Iterative feedback gradually shifts recommendations toward popular-heavy content, reducing balance between mainstream and niche exposure. These users ultimately converge toward recommendation patterns similar to G_1, reflecting difficulty in maintaining heterogeneous preferences under popularity reinforcement.*
- *Niche-focused Group (G_3): Starts with the highest APLT, LTC, and Entropy values, indicating broad and diverse exposure to tail items. However, this group experiences the sharpest proportional declines over iterations, especially under WBPR and NEUMF, showing that feedback processes gradually erode their long-tail and diversity advantages. Although G_3 retains higher absolute diversity than other groups, its relative position weakens significantly, leaving it increasingly disadvantaged in long-term iterations.*

- As an overall evaluation, niche-focused users (G_3) emerge as the most systematically disadvantaged segment. Despite being the users the system should satisfy the most, those with the largest profile sizes and highest activity levels, as shown in Table 2, feedback loops gradually erode their experience. They start with strong accuracy (*n*DCG) and broad exposure to long-tail and diverse content (APLT, LTC, Entropy), yet iterative cycles sharply diminish these advantages. Their calibration (ΔGAP) deteriorates most rapidly, their access to niche and novel items declines across all models, and their ranking accuracy decreases proportionally more than that of any other group. This convergence toward popular-heavy recommendations demonstrates that even the system's most informative users become increasingly misaligned and underserved over time, revealing that feedback-driven popularity reinforcement ultimately alienates those most critical for sustaining personalization, diversity, and long-term system value.

6 Limitations

While this study provides important insights into the dynamics of fairness, calibration, accuracy, and beyond-accuracy metrics under feedback loops, several limitations should be acknowledged.

First, the analysis is based on an offline simulation framework rather than live user interactions. Although this approach enables controlled experiments and ensures reproducibility, it may not fully capture the complexities of real-world user behavior, such as evolving preferences, context-awareness, or multi-platform interactions. As such, the external validity of the findings is limited, and results may differ in live deployment scenarios.

Second, the proposed feedback loop framework introduces a non-trivial computational cost, especially for large-scale systems with millions of users and items. Iteratively updating user profiles and recomputing recommendations across multiple metrics is resource-intensive and may limit scalability. Although feasible for research-scale datasets, additional optimization strategies or parallelization techniques will be necessary for deployment in industry-scale recommendation platforms.

Finally, while the framework considers both new and repeat consumption scenarios, the user behavior modeling remains simplified. More complex behavioral patterns (e.g., strategic avoidance of popular items, temporal shifts in interests) are not captured, which may further influence fairness outcomes. Addressing these aspects in future work will enhance the realism and robustness of the framework.

These limitations highlight the need for future research to complement simulation-based analyses with live user studies, extend scalability to large-scale systems, and refine behavior modeling. Such efforts will be crucial to translate the insights of this work into actionable strategies for fairness-aware RSs in practice.

7 Conclusion and future work

This study introduces a dynamic feedback loop framework to evaluate fairness, calibration, accuracy, and beyond-accuracy performance in recommender systems. By analyzing user groups with varying preferences on popular items (i.e., *Popular-*, *Diverse-*, and *Niche-focused*), we uncover significant disparities in how these groups are affected by feedback loops, with implications for fairness and recommendation quality.

A key contribution of the proposed framework is its ability to capture realistic user interaction dynamics. The feedback loop mechanism accounts for both new consumption, where users are exposed to and adopt novel items, and repeat consumption, where users continue engaging with previously consumed or popular content. Importantly, these dynamics are modeled in a controlled simulation environment that preserves the original structure of the datasets. This ensures that observed disparities and performance shifts emerge solely from the iterative recommendation process, rather than from alterations to the underlying data.

The results reveal persistent group-specific disparities reinforced by feedback dynamics. *Niche-focused* users, despite being the most active and information-rich segment with the largest profile sizes, experience the strongest deterioration across almost all dimensions. Their calibration worsens most rapidly, their access to long-tail and diverse items declines steadily, and their relative accuracy weakens over iterations. These findings demonstrate that feedback loops progressively misalign the system with its most valuable users. *Popular-focused* users continue to benefit the most from the system's alignment with mainstream content, achieving consistent accuracy gains but remaining confined to low diversity and zero long-tail exposure, particularly evident in the PER dataset. Meanwhile, *Diverse-focused* users, initially balanced between popular and niche preferences, gradually converge toward popular-heavy recommendation patterns, losing both calibration and diversity advantages. Collectively, these patterns confirm that iterative feedback amplifies systemic popularity bias, narrowing recommendation diversity and disproportionately disadvantaging users whose profiles contribute the richest information to personalization.

Overall, the findings emphasize that popularity bias manifests differently across groups: *Niche-focused* users remain structurally disadvantaged, *Popular-focused* users gain in accuracy but lose in diversity, and *Diverse-focused* users experience the sharpest decline over time. Future research should refine the framework by incorporating more realistic user behavior models and hybrid recommendation strategies that balance accuracy with fairness and diversity objectives. Expanding the analysis to domains such as healthcare and education, where fairness and diversity are critical, will further test the applicability and societal impact of fairness-aware recommender systems.

Acknowledgements This work is based on the Master's thesis of Yildiz Zoralioglu and builds upon the research conducted as part of her graduate studies.

Author Contributions Yildiz Zoralioglu: Software, Methodology, Visualization, Original Draft Preparation. Emre Yalcin: Methodology, Conceptualization, Investigation, Review and Editing, Supervision.

Funding Open access funding provided by the Scientific and Technological Research Council of Türkiye (TÜBİTAK). No funding was received to assist with the preparation of this manuscript.

Data Availability No datasets were generated or analysed during the current study.

Declarations

Declaration of generative AI in scientific writing In preparing this manuscript, the authors made use of ChatGPT to improve grammar, fluency, and overall readability of the text. Following this assistance, the authors carefully reviewed and revised the content, and they bear full responsibility for the final version of the publication.

Competing interests The authors declare no competing interests.

References

Abdollahpouri, H. (2020). Popularity bias in recommendation: A multi-stakeholder perspective. PhD thesis, University of Colorado at Boulder

Abdollahpouri, H., & Burke, R. (2019). Multi-stakeholder recommendation and its connection to multi-sided fairness. arXiv preprint arXiv:1907.13158

Abdollahpouri, H., Burke, R., & Mobasher, B. (2017). Controlling popularity bias in learning-to-rank recommendation. *In: Proceedings of the 11th ACM conference on recommender systems. Association for Computing Machinery, New York, NY, USA, RecSys '17* (pp. 42–46). https://doi.org/10.1145/3109859.3109912

Abdollahpouri, H., Mansoury, M., & Burke, R., et al. (2019a) The impact of popularity bias on fairness and calibration in recommendation. arXiv preprint arXiv:1910.05755

Abdollahpouri, H., Mansoury, M., & Burke, R., et al. (2019b). The unfairness of popularity bias in recommendation. arXiv preprint arXiv:1907.13286

Abdollahpouri, H., Mansoury, M., & Burke, R., et al. (2021). User-centered evaluation of popularity bias in recommender systems. *In: Proceedings of the 29th ACM conference on user modeling, adaptation and personalization* (pp. 119–129). https://doi.org/10.1145/3450613.3456821

Alizadeh Noughabi, H., Behkamal, B., Zarrinkalam, F., et al. (2025). Persuasive explanations for path reasoning recommendations. *Journal of Intelligent Information Systems, 63*(2), 413–439. https://doi.org/10.1007/s10844-024-00896-3

Bobadilla, J., Gutiérrez, A., Yera, R., et al. (2023). Creating synthetic datasets for collaborative filtering recommender systems using generative adversarial networks. *Knowledge-Based Systems, 280*, Article 111016. https://doi.org/10.1016/j.knosys.2023.111016

Bogers, T., & Van Den Bosch, A. (2009). Collaborative and content-based filtering for item recommendation on social bookmarking websites. *Iin: proceedings of the acm recsys '09 workshop on recommender systems and the social web* (pp. 9–16)

Boratto, L., Fenum G., & Marras, M., et al. (2022). Consumer fairness in recommender systems: Contextualizing definitions and mitigations. *In: Advances in information retrieval* (pp. 552–566). Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-99736-6_37

Boratto, L., Fenu, G., & Marras, M. (2021). Connecting user and item perspectives in popularity debiasing for collaborative recommendation. *Information Processing & Management, 58*(1), Article 102387. https://doi.org/10.1016/j.ipm.2020.102387

Borges, R., & Stefanidis, K. (2021). On mitigating popularity bias in recommendations via variational autoencoders. *In: Proceedings of the 36th annual ACM symposium on applied computing* (pp. 1383–1389). Association for Computing Machinery, New York, NY, USA, SAC '21. https://doi.org/10.1145/3412841.3442123

Boutilier, C., Mladenov, M., & Tennenholtz, G. (2024). Recommender ecosystems: A mechanism design perspective on holistic modeling and optimization. *In: Proceedings of the AAAI conference on artificial intelligence* (pp. 22575–22583). https://doi.org/10.1609/aaai.v38i20.30266

Burke, R. (2017). Multisided fairness for recommendation. arXiv preprint arXiv:1707.00093

Castells, P., Hurley, N., & Vargas, S. (2022). Novelty and diversity in recommender systems. *In: Recommender systems handbook* (pp. 603–646). Springer US, New York, NY. https://doi.org/10.1007/978-1-0716-2197-4_16

Chen, J., Dong, H., Wang, X., et al. (2023). Bias and debias in recommender system: A survey and future directions. *ACM Transactions on Information Systems, 41*(3), 1–39. https://doi.org/10.1145/3564284

Chen, J., Wang, C., Wang, J., et al. (2016). Recommendation for repeat consumption from user implicit feedback. *IEEE Transactions on Knowledge and Data Engineering, 28*(11), 3083–3097. https://doi.org/10.1109/TKDE.2016.2593720

Chia, P.J., Tagliabue, J., & Bianchi, F., et al. (2022). Beyond NDCG: Behavioral testing of recommender systems with reclist. *In: Companion Proceedings of the Web Conference 2022* (pp. 99–104). Association for Computing Machinery, New York, NY, USA, WWW '22. https://doi.org/10.1145/3487553.3524215

Elahi, M., Kholgh, D. K., Kiarostami, M. S., et al. (2021). Investigating the impact of recommender systems on user-based and item-based popularity bias. *Information Processing & Management, 58*(5), Article 102655. https://doi.org/10.1016/j.ipm.2021.102655

Gantner, Z., Drumond, L., & Freudenthaler, C., et al. (2012). Personalized ranking for non-uniformly sampled items. *In: Proceedings of KDD Cup 2011* (pp. 231–247). PMLR

Guan, J., Chen, B., & Yu, S. (2024). A hybrid similarity model for mitigating the cold-start problem of collaborative filtering in sparse data. *Expert Systems with Applications, 249*, Article 123700. https://doi.org/10.1016/j.eswa.2024.123700

Gulsoy, M., Yalcin, E., & Bilge, A. (2023). Robustness of privacy-preserving collaborative recommenders against popularity bias problem. *PeerJ Computer Science, 9*, Article e1438. https://doi.org/10.7717/peerj-cs.1438

Gulsoy, M., Yalcin, E., Tacli, Y., et al. (2025). Duor: Dynamic user-oriented re-ranking calibration strategy for popularity bias treatment of recommendation algorithms. *International Journal of Human-Computer Studies, 203*, Article 103578. https://doi.org/10.1016/j.ijhcs.2025.103578

Harper, F. M., & Konstan, J. A. (2015). The movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems, 5*(4), 1–19. https://doi.org/10.1145/2827872

He, X., Liao, L., & Zhang, H., et al. (2017). Neural collaborative filtering. In: Proceedings of the 26th International Conference on World Wide Web. *International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, WWW '17* (pp. 173–182). https://doi.org/10.1145/3038912.3052569

Jannach, D., Lerche, L., & Kamehkhosh, I. (2015). Beyond "hitting the hits": Generating coherent music playlist continuations with the right tracks. *In: Proceedings of the 9th ACM Conference on Recommender Systems* (pp. 187–194). Association for Computing Machinery, New York, NY, USA, RecSys '15. https://doi.org/10.1145/2792838.2800182

Klimashevskaia, A., Jannach, D., Elahi, M., et al. (2024). A survey on popularity bias in recommender systems. *User Modeling and User-Adapted Interaction, 34*(5), 1777–1834. https://doi.org/10.1007/s11257-024-09406-0

Kowald, D., Schedl, M., & Lex, E. (2020). The unfairness of popularity bias in music recommendation: A reproducibility study. *In: Advances in information retrieval* (pp 35–42). Springer International Publishing, Cham. https://doi.org/10.1007/978-3-030-45442-5_5

Lesota, O., Melchiorre, A., & Rekabsaz, N., et al. (2021). Analyzing item popularity bias of music recommender systems: are different genders equally affected? *In: Proceedings of the 15th ACM conference on recommender systems* (pp. 601–606). https://doi.org/10.1145/3460231.3478843

Li, R.Z. (2023). Metric optimization and mainstream bias mitigation in recommender systems. arXiv preprint arXiv:2311.06689

Liang, D., Krishnan, R.G., & Hoffman, M.D., et al. (2018). Variational autoencoders for collaborative filtering. *In: Proceedings of the 2018 world wide web conference* (pp. 689–698). International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, WWW '18. https://doi.org/10.1145/3178876.3186150

Lin, A., Wang, J., & Zhu, Z., et al. (2022). Quantifying and mitigating popularity bias in conversational recommender systems. *In: Proceedings of the 31st ACM international conference on information & knowledge management* (pp. 1238–1247). Association for Computing Machinery, New York, NY, USA, CIKM '22. https://doi.org/10.1145/3511808.3557423

Lü, L., Medo, M., Yeung, C. H., et al. (2012). Recommender systems. *Physics Reports, 519*(1), 1–49. https://doi.org/10.1016/j.physrep.2012.02.006

Mansoury, M., Abdollahpouri, H., & Pechenizkiy, M., et al. (2020). Feedback loop and bias amplification in recommender systems. in: proceedings of the 29th acm international conference on information & knowledge management. (pp. 2145–2148). Association for Computing Machinery, New York, NY, USA, CIKM '20. https://doi.org/10.1145/3340531.3412152

Marcuzzo, M., Zangari, A., Albarelli, A., et al. (2022). Recommendation systems: An insight into current development and future research challenges. *IEEE Access, 10*, 86578–86623. https://doi.org/10.1109/ACCESS.2022.3194536

McNee, S.M., Riedl, J., & Konstan, J.A. (2006). Being accurate is not enough: How accuracy metrics have hurt recommender systems. *In: CHI '06 extended abstracts on human factors in computing systems*, (pp. 1097–1101). Association for Computing Machinery, New York, NY, USA, CHI EA '06. https://doi.org/10.1145/1125451.1125659

Mladenov, M., Hsu, C.W., & Jain, V., et al. (2020). Demonstrating principled uncertainty modeling for recommender ecosystems with recsim ng. *In: Proceedings of the 14th ACM conference on recommender systems* (pp. 591–593). Association for Computing Machinery, New York, NY, USA, RecSys '20. https://doi.org/10.1145/3383313.3411527

Nguyen, T. T., Maxwell Harper, F., Terveen, L., et al. (2018). User personality and user satisfaction with recommender systems. *Information Systems Frontiers, 20*, 1173–1189. https://doi.org/10.1007/s10796-017-9782-y

Salah, A., Truong, Q. T., & Lauw, H. W. (2020). Cornac: A comparative framework for multimodal recommender systems. *Journal of Machine Learning Research, 21*(95), 1–5.

Sanders, R. (1987). The pareto principle: its use and abuse. *Journal of Services Marketing, 1*(2), 37–40. https://doi.org/10.1108/eb024706

Sharma, R. S., Shaikh, A. A., & Li, E. (2021). Designing recommendation or suggestion systems: looking to the future. *Electronic Markets, 31*(2), 243–252. https://doi.org/10.1007/s12525-021-00478-z

Steck, H. (2018). Calibrated recommendations. *In: Proceedings of the 12th ACM conference on recommender systems* (pp. 154–162). Association for Computing Machinery, New York, NY, USA, RecSys '18. https://doi.org/10.1145/3240323.3240372

Suhaim, A. B., & Berri, J. (2021). Context-aware recommender systems for social networks: Review, challenges and opportunities. *IEEE Access, 9*, 57440–57463. https://doi.org/10.1109/ACCESS.2021.3072165

Vargas, S., & Castells, P. (2011). Rank and relevance in novelty and diversity metrics for recommender systems. *In: Proceedings of the Fifth ACM conference on recommender systems* (pp. 109–116). Association for Computing Machinery, New York, NY, USA, RecSys '11. https://doi.org/10.1145/2043932.2043955

Vercoutere, S., De Pessemier, T., & Martens, L. (2025). Hybrid transformer-based recommender system for political news. *Journal of Intelligent Information Systems, 63*, 1569–1601. https://doi.org/10.1007/s10844-025-00951-7

Wang, S., Gong, M., Li, H., et al. (2016). Multi-objective optimization for long tail recommendation. *Knowledge-Based Systems, 104*, 145–155. https://doi.org/10.1016/j.knosys.2016.04.018

Waris, M., Zaman Fakhar, M., Gulsoy, M., et al. (2024). A novel pre-processing technique to combat popularity bias in personality-aware recommender systems. *IEEE Access, 12*, 183230–183251. https://doi.org/10.1109/ACCESS.2024.3510475

Wei, F., & Chen, S. (2025). Multi-view collaborative training and self-supervised learning for group recommendation. *Mathematics, 13*(1), 66. https://doi.org/10.3390/math13010066

Yalcin E (2022). Pophybrid: a novel item popularity-aware hybrid approach for long-tail recommendation. *In: 2022 International congress on human-computer interaction, optimization and robotic applications (HORA)* (pp. 1–6). IEEE. https://doi.org/10.1109/HORA55278.2022.9800006

Yalcin, E., & Bilge, A. (2022). Evaluating unfairness of popularity bias in recommender systems: A comprehensive user-centric analysis. *Information Processing & Management, 59*(6), Article 103100. https://doi.org/10.1016/j.ipm.2022.103100

Yalcin, E., & Bilge, A. (2022). Treating adverse effects of blockbuster bias on beyond-accuracy quality of personalized recommendations. *Engineering Science and Technology, an International Journal, 33*, Article 101083. https://doi.org/10.1016/j.jestch.2021.101083

Yalcin, E., & Bilge, A. (2023). Popularity bias in personality perspective: An analysis of how personality traits expose individuals to the unfair recommendation. *Concurrency and Computation: Practice and Experience, 35*(9), Article e7647. https://doi.org/10.1002/cpe.7647

Yao, F., Li, C., & Nekipelov, D., et al. (2023). How bad is top-k recommendation under competing content creators? *In: International conference on machine learning* (pp. 39674–39701). PMLR

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Authors and Affiliations

Yildiz Zoralioglu[1] · Emre Yalcin[2]

✉ Emre Yalcin
eyalcin@cumhuriyet.edu.tr

Yildiz Zoralioglu
yildizzoralioglu@gmail.com

[1] Graduate School of Natural and Applied Sciences, Sivas Cumhuriyet University, Sivas 58140, Turkey

[2] Computer Engineering Department, Sivas Cumhuriyet University, Sivas 58140, Turkey