Enhancing Calibration and Reducing Popularity Bias in Recommender Systems[⋆]

Rodrigo Ferrari de Souza[1][0000−0002−9272−4107] and Marcelo Garcia Manzato[1][0000−0003−3215−6918]

Mathematics and Computer Science Institute, University of São Paulo, Av. Trab. Sancarlense 400, São Carlos-SP, Brazil
rodrigofsouza@usp.br, mmanzato@icmc.usp.br

Abstract. The recent literature highlights that recommendation systems are significantly influenced by popularity bias. This phenomenon has far-reaching implications for the fairness and accuracy of recommendations. This bias often results in some users finding their preferences inadequately reflected in their recommendations, while others benefit from more consistent suggestions. Nevertheless, despite the current state-of-art efforts in this field that primarily aim to provide fairer recommendations, a crucial aspect has been overlooked: the impact of popularity bias on the long tail effect, which leads to a decline in the visibility of less popular items in recommendations. To address this research gap, the present study introduces a calibration approach designed to cater to the diverse interests of users across various levels of item popularity. To achieve this objective, we propose a post-processing system that is independent of any specific recommendation algorithm. Building upon the foundational idea presented by [20], we evaluate the efficacy of our proposed system using an additional dataset from the domain of music. The performance assessment of our system encompasses a range of metrics that consider aspects related to popularity, accuracy, and fairness. Additionally, four recommendation algorithms and two distinct baselines are employed. As a result, the proposed technique mitigates popularity bias, augmenting diversity and fairness within the considered datasets.

Keywords: Recommender System · Popularity Bias · Fairness · Calibration.

1 Introduction

Machine learning algorithms have gained significant prominence in people's lives from various application domains. This prevalence has empowered researchers to gain insights into prevalent issues existing within these systems, such as recommendation imbalances [18]. This specific problem entails certain items being prioritized over others in recommendation processes, leading to an inherently unfair and biased system.

[⋆] Supported by FAPESP, process number 2022/07016-9

The unfairness present in recommendation systems stems from various types of biases that naturally occur in the data. One of these biases is popularity bias, which tends to favor the recommendation of highly popular items at the cost of lesser-known ones [8]. This outcome leads to a situation where users may overlook certain items or the system struggles to provide recommendations that align with users' preferences.

In this context, the notion of fairness within a recommendation system is tied to its ability to provide recommendations that align consistently with the preferences of all users. Therefore, it is important to measure the system's capability to offer appropriate recommendations to any user, and one promising approach is employing calibration [22]. Calibration has gained renewed attention recently, particularly with studies about fairness in machine learning algorithms. A calibrated algorithm is when the predicted proportions of different classes correspond to the factual ratios of instances within the available data.

A specific type of calibration is when using items' metadata (e.g. genres) to provide a recommendation ranking whose distribution of categories is aligned with the distribution of these topics in the user's profile. In this scenario, [22] introduces a system that adjusts the recommended items ratio based on users' proportion interest in this aspect. Meanwhile, [10] and [21] have focused on strategies to align recommendations consistently and with high accuracy, ensuring the inclusion of items tailored to users' preferences. In an attempt to mitigate the disparity in recommendations based on users' gender and age, [11] introduces an approach to reclassify results considering these attributes.

Although these works address fairness regarding item categories, they are prone to popularity bias. Other works, such as [28, 16, 9, 19, 15, 30], focus solely on reducing popularity bias but do not address fairness regarding categories.

Hence, there exists a gap in the field for systems that address popularity bias while also being calibrated to respect users' preference levels. In a previous work [20], we proposed a personalized calibration technique, which uses popularity and genre calibrations in a switch-based approach to provide fairer recommendations to users according to their interests. We showed that calibrating items based on popularity is a way to improve a recommendation system to bring fairer recommendations to users to meet their preferences and reduce the impact of popularity bias in the system. However, the system was evaluated with two datasets from a single domain – the movies domain – and analyzing the system's behavior within other contexts was left to future work.

In this work, we propose a deeper analysis of our calibration method, which is performed with an additional dataset from another domain. We also include two different recommendation algorithms to verify the performance of its calibration and two additional metrics that favor the analysis in terms of calibration and fairness. As a result, we use two datasets from the movie domain (MovieLens-20M and Yahoo Movies) and one from the music domain (Yahoo Songs). Additionally, we select four traditional recommendation algorithms (SVD++, NMF, Item KNN, and SlopeOne) calibrated with our approach and with two baselines.

The structure of this paper is outlined as follows. Section 2 presents related works and compares them to our proposal. Section 3 defines the problem and our approach. Section 4 elaborates on the experimental setup, and Section 5 provides a result analysis for the three evaluated datasets. Finally, in Section 6, we conclude the study by highlighting promising results compared to state-of-the-art and suggesting potential avenues for future research.

2 Related Work

In the literature, for a recommendation list to be calibrated, it is necessary that an equivalent number of items be returned for each user in relation to the topics the user has previously appreciated [21]. In [21], the authors present a strategy for calibrating recommendation systems to balance accuracy and consistency. However, their approach is heuristic-based and does not effectively address the issue of popularity bias.

[12] adopt a calibration strategy inspired by [22]. They replace the genre-based calibration of items with a sub-profiles approach focused on users' interests. [32] demonstrate that systems that adopt Bayesian Personalized Ranking make unfair recommendations, as they favor items over others. The work then proposes a calibration model capable of reducing unfairness. These works differ from ours, as we focus on each user's interest level by popularity.

Further contributing to the discourse on handling user preferences in relation to popularity, [2] proposes an approach that categorizes users into distinct preference groups based on their interest in popular items. We also adopt this preference categorization method in our work, albeit employing a distinct calibration methodology. We introduce a genre-based calibration to tailor recommendations for users favoring less popular items, ultimately enhancing engagement. We compared the performance of the two systems in our work.

In an approach adopted by [29, 6], the system reclassifies the recommended items penalizing the most popular items, reducing bias and accuracy. This approach is different from our work, as in our study, we used the popularity aspect to rank items according to the user's preferences.

In pursuing fairness, [4] introduces metrics to gauge recommendation system fairness and proposes a pairwise regularization technique to enhance fairness during algorithm training. Despite its promising results, this work diverges from ours as it is not a post-processing step and does not consider assessing user interest levels regarding popularity bias.

Other notable calibration approaches to mitigate biases include [25]. This work delves into modeling the causal effects on user representation during item score prediction, which is suitable for various recommendation algorithms. While this approach differs from ours, it aligns with the goal of reducing bias.

There is also an approach that adopts Inverse Propensity Weighting (IPW), where the impact of popular items is reduced in the training phase through the analysis of a cause-and-effect relationship to reduce the problem of popularity bias [26]. This method is compatible with many models, as is our proposal, but it is applied in the training phase. [5] calibrate the system based on the long tail to measure how the system treats its items equally in this distribution, proposing a metric to minimize the biased correlation between the item and its popularity. [31] attempt to remove the bias at the same moment recommendations are generated, unlike our approach, whose bias removal is accomplished in a post-processing step. The performance of these last three works [26, 5, 31] were already compared against our proposal in our previous work [20], and due to their low performance, in particular with fairness metrics, were not included in the current paper's analysis.

Compared to the findings of [20], in this paper, we conducted our experiments on an additional dataset, the Yahoo Songs, introducing a shift in the domain for recommendation and calibration. As our proposal is independent of the recommendation algorithm, we also included other traditional recommenders to verify how calibration is performed with these approaches. These extensions allow us to explore the robustness of the system's performance regarding fairness when exposed to varying contexts.

3 Problem and Approach

This section formalizes the problem we are dealing with and the proposed solution.

Problem Statement. To formalize the problem, we adhere to the framework proposed by [20]. Suppose we have a set of items $I = \{i_1, i_2, ..., i_{|I|}\}$, a set of users $U = \{u_1, u_2, ..., u_{|U|}\}$ and a set of candidate items for each user $CI_u = \{i_1, i_2, ..., i_N\}$, where N is the number of items suggested by the recommender system. We have two valuable pieces of information to know the user's preferences: (1) genres or metadata of items interacted by user u; and (2) popularity of items interacted by user u. Our task is to exploit users' preferences related to popularity and genres to increase the fairness and accuracy of recommendations and reduce popularity bias by increasing the long tail coverage of all users based on CI_u generated by any recommendation model.

Approach. We propose the personalized calibration approach as shown in Figure 1. In particular, our approach is divided into two methods: **popularity calibration** (highlighted in continuous blue line in Figure 1), which extends the genre calibration previously proposed by [22] (highlighted in the dotted red line in Figure 1); and **personalized calibration** (Figure 1 as a whole), which uses genre calibration and popularity calibration in a unified model to provide recommendations calibrated according to popularity and genres.

To calibrate the recommendation list based on the popularity of the items consumed by the user in the past, we introduce a popularity division to group items based on how much users access them. In addition, to provide personalized calibration based on popularity and genres, we first group users based on their consumption. In this way, if the user consumes popular items below an established limit, we perform a genre calibration; otherwise, we perform a popularity calibration to meet the preference level for this aspect.

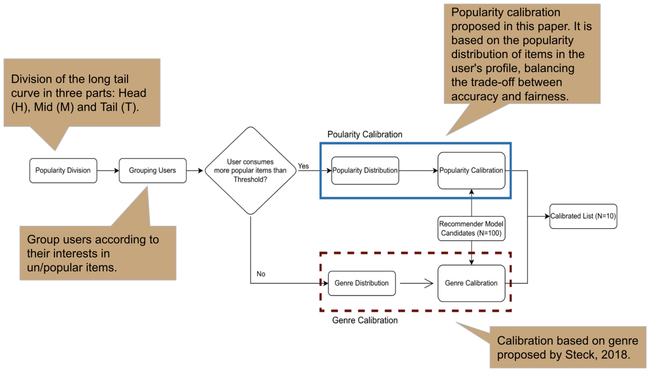
Popularity calibration proposed in this paper. It is based on the popularity distribution of items in the user's profile, balancing the trade-off between accuracy and fairness.
Division of the long tail curve in three parts: Head (H), Mid (M) and Tail (T).
Poularity Calibration
Popularity Division
Grouping Users
User consumes more popular items than Threshold?
Yes
Popularity Distribution
Popularity Calibration
Calibrated List (N=10)
Recommender Model Candidates (N=100)
Group users according to their interests in un/popular items.
No
Genre Distribution
Genre Calibration
Genre Calibration
Calibration based on genre proposed by Steck, 2018.

Fig. 1. Flowchart showing our personalized method highlighting the calibration steps

3.1 Popularity Division

The popularity division, introduced in [20] and shown in Figure 1, is based on recommender systems' long tail concept. We propose to divide this curve into three parts. The **Head (H)**, with items representing the top 20% of the total of past interactions. Then, we get the **Tail (T)** with items that sum the less 20% of interactions. Finally, the **Mid (M)** group contains items that are neither **Head (H)** nor **Tail (T)**. It is worth mentioning that this division by percentage was chosen based on Pareto's principle.

3.2 Grouping Users

As shown in Figure 1, our unified model switches between popularity and genre calibration. To make this decision, it is required to group users according to their interests in unpopular/popular items. So, we defined the threshold as the mean of all ratios, which is a value that can be easily computed on every dataset, as shown in Equation 1:

$$G_{threshold} = \frac{\sum_{u}^{U} \frac{\sum_{i}^{I_u} \mathbf{1}(i)}{|I_u|}}{|U|} \tag{1}$$

where $\mathbf{1}(i)$ is an indicator function that returns 1 if the item i, interacted by user u, is in the **H** popularity category. Finally, we assume that if the ratio of items in the category **H** is lower than $G_{threshold}$, then we should get a recommendation list calibrated by genre; otherwise, by popularity.

3.3 Popularity Distributions

In this work, we adapted the formulation proposed by [22] to calibrate recommendation lists based on the item's popularity. His work assumes items can have more than one genre, which is not valid in our context, where an item has only a level of popularity. So, instead, we calculate the sums of weights of every popularity type over the sum of all weights.

The $p(t|u)$ is defined as the target distribution based on the popularity of the items the user interacted with in the past. In Equation 2 we used the weights $w_r(u,i)$ as the explicit or implicit rating the user u gave to the item i:

$$p(t|u) = \frac{\sum_{i \in I_u} w_r(u,i) \cdot p(t|i)}{\sum_{i \in I_u} w_r(u,i)} \tag{2}$$

where $p(t|i)$ is defined as 1 if the item i is in the popularity category t. Then to deal with the recommended list distribution, Equation 3 defines $q(t|u)$:

$$q(t|u) = \frac{\sum_{i \in R_u^*} w_p(u,i) \cdot p(t|i)}{\sum_{i \in R_u^*} w_p(u,i)} \tag{3}$$

In this case, we use the weights $w_p(u,i)$ as the rank position of the item i in the reordered recommended list to the user u.

3.4 Fairness Measure

In our context, the system is fair when it meets the popularity proportions expected by the user. Therefore, if the user consumes fewer popular items than the established limit, we will perform a genre-based calibration because, in this case, the user does not care about the popularity of the items. If the user consumes more popular items than the established limit, we will calibrate based on popularity, respecting the user's level of interest in this aspect. Several metrics assess fairness in recommender systems [24]. However, in our case, we use the Kullback-Leibler for the same reasons pointed out by [22] and exploited by [10].

The Kullback-Leibler quantifies the inequality in the interval $[0,\infty]$, where 0 means both distributions are almost the same, and higher values indicate unfairness. Also, we adopted the regularization proposed by [22], which defined the $\alpha = 0.01$ as a regularization variable to avoid zero division when $q(t|u)$ goes to zero.

$$D_{KL}(p \| q) = \sum_t p(t|u) \cdot log \frac{p(t|u)}{(1-\alpha) \cdot q(t|u) + \alpha \cdot p(t|u)} \tag{4}$$

Although there are other divergence metrics, like Hellinger and Person Chi-Square, proposed by [7] and exploited by [10], we use only the Kullback-Leibler due to its simplicity.

3.5 Calibration

We call **calibration** refereeing as the process to find the optimal set R_u^*, using the maximum marginal relevance, as shown in Equation 5, where D_{KL} is the fairness function. In this formulation, when $\lambda = 0$, it focuses only on the recommendation scores, and when $\lambda = 1$, we focus on fair items concerning the user's profile. Figure 1 shows the final calibration process.

$$R_u^* = \max_{CI_u} (1 - \lambda) \cdot \sum_{i \in CI_u} wr_{u,i} - \lambda \cdot D_{KL}(p, q(CI_u)) \tag{5}$$

It is worth mentioning that such formulation is similar to the calibration approach proposed by [22]. Consequently, although we focus on popularity in this work, it is possible to adopt a greedy approach to solve Equation 5, whose details can be found in [22].

4 Experimental Setup

This section describes the steps to reproduce our comparisons, including dataset pre-processing, baseline parameters, training, and evaluation methodology.

4.1 Datasets

In this study, we used two datasets from the movies domain and one dataset from the music domain:

- **Yahoo Movies**[1]: This dataset is a user-movie rating, where the user gives ratings from one to five to the movies they watched. In the pre-processing step, we only removed movies with no genres in the metadata. Instead of binarizing the rating as done by [22], we used the explicit feedback as the weight $w_r(u, i)$ in Equation 2.
- **MovieLens-20M**[2]: In this dataset, similar to [22] and in contrast to the Yahoo Movies dataset, we binarized the ratings by retaining interactions where the rating was greater than 4. Furthermore, due to hardware limitations, we reduced the dataset size by removing movies with fewer than ten interactions and users with fewer than 180 movies.
- **Yahoo Songs**[3]: This dataset is a user-song rating, where the user gives a rating from one to five to the songs they listened to. In the pre-processing step, we removed songs with no genres in the metadata. Due to hardware limitations, we downsized the dataset by excluding songs with less than 10 interactions and users with less than 10 rated songs.

Table 1 summarizes important statistics about the processed datasets.

[1] https://webscope.sandbox.yahoo.com/

[2] https://grouplens.org/datasets/movielens/20m/

[3] https://webscope.sandbox.yahoo.com/

Table 1. Statistics of the datasets after all pre-processing steps.

Dataset	# Users	# Ratings	# Items
Yahoo Movies	7,642	211,231	11,916
MovieLens 20M	12,603	3,984,599	10,417
Yahoo Songs	2,817	680,460	22,196

4.2 Metrics

In our experiments, we evaluated the effects of different calibration approaches in terms of precision, fairness, and popularity bias, as detailed next:

- **Precision and Quality**: we used the Mean Reciprocal Rank (MRR) and Mean Average Precision (MAP) metrics to measure the rank quality of the item in the re-ranked list. MAP and MRR range in the interval $[0, 1]$ where **higher values are better**.
- **Fairness**: we used a metric proposed by [10], called Mean Rank Miscalibration (MRMC), which covers the interval $[0, 1]$, where **lower values are better**. Initially, it was used to compute the fairness in genres on the recommendation list, but we also used it to calculate the popularity miscalibration in our work. As we are focusing on two types of fairness (genre and popularity), we propose in this paper to use the harmonic mean (or F1 score) between MRMC of genres and popularity, where **higher values are better**:
$$F1 = 2\frac{(1 - MRMC\ Genre) * (1 - MRMC\ Pop)}{(1 - MRMC\ Genre) + (1 - MRMC\ Pop)} \tag{6}$$
- **Popularity Bias**: we used the long-tail coverage (LTC) [3] and group average popularity (ΔGAP) [1] metrics to measure popularity bias. The LTC metric indicates the fraction of items that users see in the recommendation lists and varies in the interval $[0, 1]$, where 0 means all recommended items are the most popular and 1 means all items recommended to a user are in the less popular categories. Thus, the closer to 1, the more diverse content will be recommended [3]. The ΔGAP ranges in the interval $[-1, 1]$, where negative values mean recommendations are less popular than expected by the users' preferences, and positive values mean recommendations are more popular than expected. We also adopted three divisions of user groups, based on [1] for the ΔGAP: **BlockBuster (BB)** whose users' consumption is at least 50% of the most popular items, **Niche (N)** where users' consumption is at least 50% of the lowest popularity items and **Diverse (D)** whose users' preferences diverge from the other two groups. Finally, as optimal values of ΔGAP should be close to zero, we propose in this paper to use the Root Mean Squared Error (RMSE) among the three groups of users, where **lower values are better**:

$$RMSE = \frac{\sqrt{\Delta GAP_{BB}^2 + \Delta GAP_N^2 + \Delta GAP_D^2}}{3} \tag{7}$$

4.3 Evaluation Methodology

We executed three times the calibration process in the experiments involving the MovieLens, Yahoo Songs, and Yahoo Movies datasets. We got the mean of the values outputted by the metrics to guarantee the stability of the results. Also, the train and test sets were chosen by randomly splitting the dataset in 70/30% of interactions, respectively [2, 10].

The process of calibration does not depend on the recommender system algorithm. It acts as a post-processing step where, after the model predicts the candidate items for a user, we apply the calibration technique described in Equation 5 to find the best list of items for that user. Consequently, to understand how the calibration approaches perform under different recommender algorithms, we used four well-known models described below, based on [22] and [10] works. For some models, we used the implementation provided by [23].

1. **SVD++**: Singular Value Decomposition extension [13] to work with implicit feedback. Similarly to [10], we used $ne = 20$ as the number of epochs, $\gamma_u = \gamma_i = 0.005$ as the learning rate for users and items, $\lambda_u = \lambda_i = 0.02$ as regularization constants, and $f = 20$ factors.
2. **NMF**: Non-negative Matrix Factorization proposed by [17]. Similarly to [10], we used $ne = 50$, $\gamma_u = \gamma_i = 0.005$, $\lambda_u = \lambda_i = 0.06$ and $f = 15$.
3. **Item KNN**: To implement this collaborative filtering algorithm, we employed the KNNWithMeans approach from [23], using $k = 30$ nearest neighbors. Additionally, we specified the Pearson correlation coefficient as the metric for item similarity.
4. **SlopeOne**: A collaborative filtering algorithm, whose implementation was based on the Surprise library [23] with default values for the parameters.

The experiment for each recommender system consists of using the training data to feed the model to learn the user's preferences based on the items interacted in the past, represented as I_u. After the training step, we predict all missing ratings, and for every user, we select the top 100 items with the highest predicted rating, represented as CI_u. We use the weight $w_r(u, i)$ as the rating the algorithm predicted for the candidate item. Finally, the final recommendation list R_u^* is created with the top 10 items given by the calibration process.

Regarding our proposal, we separately analyze the performance of **popularity calibration** and **personalized calibration**. Both calibrations were described in Section 3. For the trade-off between similarity and fairness metrics, in Equation 5, we adopted the values described by [22], ranging from $\lambda \in [0, 0.1, 0.2, \ldots, 1]$.

4.4 Baselines

To compare the efficiency of our proposed method regarding the popularity bias, we selected two state-of-the-art methods specialized in popularity debiasing and genre calibration. To make a fair comparison, we applied the same train-test split methodology and result stability to calibration approaches.

1. **Genres**: Proposed by [22], this method implements a calibration technique for genres. We followed the authors' implementation, and this method is compared against our proposals using the same set of four recommender algorithms (SVD++, NMF, ItemKNN and SlopeOne).
2. **CP**: Proposed by [2], this method implements a calibration technique for popularity, similar to our proposed popularity calibration, but using the Jensen-Shannon divergence metric for comparing the profile and recommendation distributions. We followed the authors' method for both datasets to split the popularity into groups and exploited the parameter $\lambda \in [0, 1]$. This method is compared against our proposals using the same set of four recommender algorithms.

Our evaluation consists of four recommenders, four calibration approaches (two proposals and two baselines), and eleven trade-off weights, resulting in $4 \times 11 \times 4 = 176$ combinations of the recommendation list to be evaluated for each dataset.

5 Results

This section presents the results of our experiments for all datasets: Yahoo Movies, Yahoo Songs, and MovieLens 20M.

Each section presents the results for Mean Reciprocal Rank, Genre Mean Rank Miscalibration, Popularity Mean Rank Miscalibration, Long Tail Coverage, and a comparison against the baselines.

All approaches (proposal and baselines) have a varying number of trade-off values (λ in Equation 5). We varied this parameter from 0.0 to 1.0, as shown in the following figures, and then, for the comparison against baselines, we selected the trade-off value that achieved the highest LTC value.

5.1 Yahoo Movies

Mean Reciprocal Rank. Figure 2 shows the MRR results on the Yahoo Movies dataset. We observe that, except for the SlopeOne recommender, for all other recommenders, at least one of our proposed methods overcomes the two baselines (CP and Genres). We also notice that with an increase in the trade-off (λ parameter in Equation 5), the MRR tends to achieve higher values than the MRR obtained in a recommendation list without any calibration ($\lambda = 0$). In particular, the NMF model performed best compared to other recommenders. Indeed, it benefited most from the calibration approaches, particularly our calibration technique based on popularity.

Genre Mean Rank Miscalibration. Figure 3 shows the results of MRMC related to genres on the Yahoo Movies dataset. Notably, when $\lambda \geq 0.1$, all methods increased the fairness related to genres, whereas calibration by genre solely performed the best in all recommenders as it was initially designed to provide fairness according to the genres.

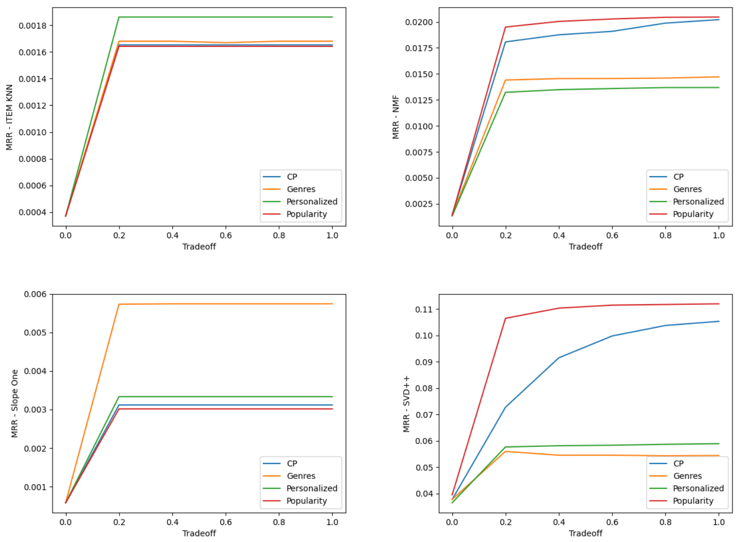

Fig. 2. MRR results over the selected recommenders and three types of calibration on the Yahoo Movies dataset.

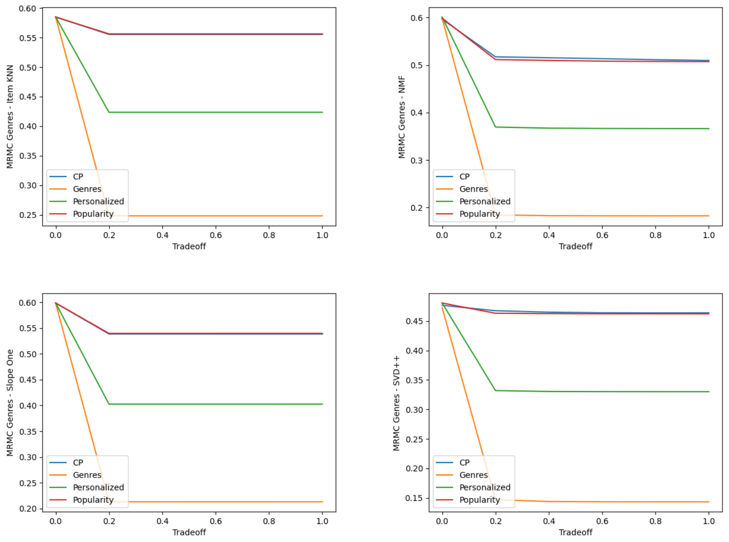

Fig. 3. MRMC of genres results over the selected recommenders and three types of calibration on the Yahoo Movies dataset.

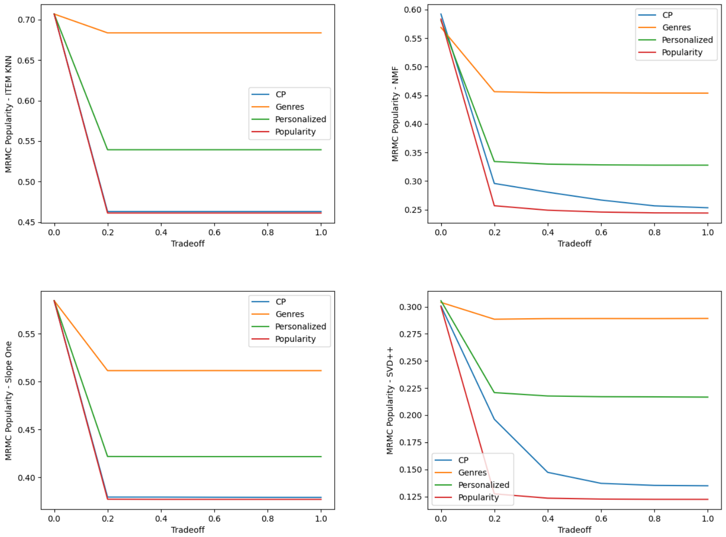

Fig. 4. MRMC of popularity results over the selected recommenders and three types of calibration on the Yahoo Movies dataset.

Popularity Mean Rank Miscalibration. Figure 4 shows the results of MRMC related to popularity on the Yahoo Movies dataset. Our proposed method, popularity calibration, improved fairness associated with popularity in all recommenders, including the CP calibration method.

Long Tail Coverage. Figure 5 shows the results of LTC on the Yahoo Movies dataset. We observed that genre and personalized calibrations were the best performers in increasing the discovery of less popular items in all recommenders. It demonstrates that, although we achieved better results on MRR, as shown in Figure 2, our personalized calibration was still able to provide diversity by covering the long tail as much as genre calibration. In turn, although it achieved better results on MRR, popularity calibration performed worse on the long tail coverage, as it does not consider the genres for calibrating the recommendation list.

Baselines Comparison. Table 2 compares our proposed personalized and popularity calibration methods against the two methods described in Subsection 4.4. For each combination between recommender and calibration, the table also shows the selected trade-off weight λ according to the best LTC value, as previously explained.

Comparing our personalized calibration proposal with CP calibration, it is possible to observe that we achieved better values for the LTC and F1 Score in

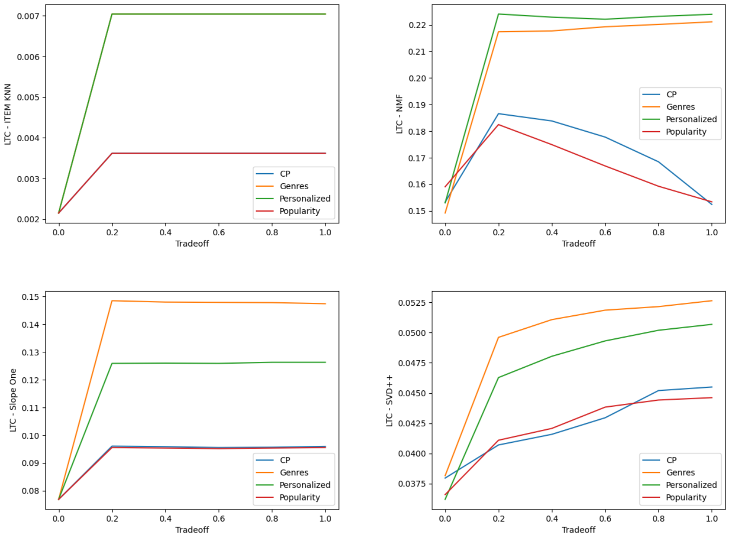

Fig. 5. LTC results over the selected recommenders and three types of calibration on the Yahoo Movies dataset.

all cases, indicating a more diverse and calibrated list according to the user's preferences. Regarding fairness among the three user groups, we note that we were able to achieve better RMMSE in two out of four recommenders. When personalized calibration was combined with the ItemKNN and SlopeOne algorithms, it yielded superior results for the MAP and MRR metrics, implying higher accuracy. However, the same was not true when using the NMF and SVD++ algorithms.

Comparing personalized calibration with genre-based calibration, we were able to achieve better LTC and F1 for the NMF and Item KNN recommenders, respectively. In addition, the accuracy of Item KNN and SVD++ was improved, as shown by the MRR and MAP metrics. Finally, we note the better fairness among the user groups for all recommenders, except for SlopeOne, whose RMSE was lower when calibration by genres was applied.

Analyzing our popularity-based calibration proposal alongside CP calibration, we noticed better F1 in all cases, indicating better calibration of genres and popularity. The proposed popularity-based calibration was also able to improve fairness in all recommenders, as shown by the lowest values in RMSE. When compared to genre-based calibration, the popularity-based was able to improve Item KNN in terms of F1 score, and Item KNN, NMF and SVD++ in terms of MRR, MAP and RMSE.

Table 2. Comparison of our proposed calibration approaches against the baselines in the Yahoo Movies dataset. The ▲ symbol means a statistically significant improvement of the proposed approach in comparison to baselines, with a p-value < 0.05 using the Student's t-test; the ● symbol denotes no statistically significant gain or loss; and the ▼ symbol indicates that the baseline is statistically better than our proposal. Each pair of symbols is related to the CP and Genres baselines, respectively.

Algorithm	**LTC**	**MRMC Genres**	**MRMC Pop.**	**F1 Score**	**MRR**	**MAP**	ΔGAP_{BB}	ΔGAP_{N}	ΔGAP_{D}	**RMSE**
Item KNN + CP ($\lambda = 1.0$)	0.004	0.555	0.463	0.486	0.002	0.001	-0.990	-0.897	-0.974	0.551
NMF + CP ($\lambda = 0.2$)	0.187	0.517	0.296	0.573	0.018	0.006	-0.941	-0.769	-0.862	0.498
SlopeOne + CP ($\lambda = 0.2$)	0.096	0.538	0.379	0.529	0.003	0.001	-0.969	-0.844	-0.926	0.528
SVD++ + CP ($\lambda = 1.0$)	0.045	0.464	0.135	0.637	0.105	0.039	-0.746	-0.126	-0.487	0.353
Item KNN + Genres ($\lambda = 1.0$)	0.007	0.248	0.684	0.445	0.002	0.001	-0.994	-0.932	-0.987	0.561
NMF + Genres ($\lambda = 1.0$)	0.221	0.182	0.454	0.655	0.015	0.005	-0.974	-0.748	-0.934	0.514
SlopeOne + Genres ($\lambda = 1.0$)	0.149	0.213	0.511	0.603	0.006	0.002	-0.982	-0.765	-0.958	0.524
SVD++ + Genres ($\lambda = 0.2$) [22]	0.053	0.143	0.289	0.777	0.054	0.019	-0.887	0.205	-0.731	0.388
Item KNN + Personalized ($\lambda = 1.0$)	0.007 ▲ ●	0.423	0.539	0.512 ▲ ▲	0.002 ▲ ▲	0.001 ▲ ▲	-0.994	-0.895	-0.978	0.552 ▼ ▲
NMF + Personalized ($\lambda = 1.0$)	0.224 ▲ ▲	0.366	0.328	0.653 ▲ ▼	0.014 ▼ ▼	0.005 ▼ ●	-0.973	-0.708	-0.873	0.494 ▲ ▲
SlopeOne + Personalized ($\lambda = 1.0$)	0.126 ▲ ▼	0.402	0.422	0.588 ▲ ▼	0.003 ▲ ▼	0.001 ▲ ▼	-0.982	-0.837	-0.936	0.532 ▼ ▼
SVD++ + Personalized ($\lambda = 1.0$)	0.051 ▲ ▼	0.330	0.217	0.722 ▲ ▼	0.059 ▼ ●	0.023 ▼ ▲	-0.882	-0.028	-0.589	0.349 ▲ ▲
Item KNN + Popularity ($\lambda = 1.0$)	0.004 ● ▼	0.556	0.461	0.487 ▲ ▲	0.002 ▲ ▲	0.001 ▲ ▲	-0.990	-0.895	-0.974	0.551 ▲ ▲
NMF + Popularity ($\lambda = 0.2$)	0.200 ▲ ▼	0.512	0.257	0.589 ▲ ▼	0.019 ● ▲	0.007 ▲ ▲	-0.934	-0.723	-0.846	0.485 ▲ ▲
SlopeOne + Popularity ($\lambda = 1.0$)	0.096 ▲ ▼	0.540	0.377	0.529 ▲ ▼	0.003 ● ▼	0.001 ▲ ▼	-0.969	-0.840	-0.926	0.527 ▲ ▼
SVD++ + Popularity ($\lambda = 1.0$)	0.045 ● ▼	0.462	0.122	0.667 ▲ ▼	0.112 ▲ ▲	0.043 ▲ ▲	-0.720	-0.037	-0.442	0.281 ▲ ▲

5.2 MovieLens 20M

Mean Reciprocal Rank. Figure 6 shows the results of MRR in the MovieLens 20M dataset. According to the trade-off values, our popularity-based calibration method achieved the best MRR in all recommenders, indicating better accuracy.

Genre Mean Rank Miscalibration. Figure 7 shows the results of MRMC related to genres in the MovieLens 20M dataset. Our methods could not achieve the best values compared to genre calibration. However, they could still increase the fairness of genres compared to the CP method.

Popularity Mean Rank Miscalibration. Figure 8 shows the results of MRMC related to popularity. Our proposed popularity calibration achieved the best results for all recommenders, while the personalized calibration outperformed the genre calibration in all scenarios.

Long Tail Coverage. Figure 9 shows the results of LTC on the MovieLens 20M dataset. It is possible to note a significant increase in Item KNN, NMF and SVD++ associated with personalized calibration. The popularity-based calibration obtained the lowest values, indicating less diversity, but in compensation, achieved the best accuracy as shown in Figure 6, following the inverse relationship between diversity and precision [14].

Baselines Comparison. Table 3 compares our proposed personalized and popularity calibration methods against the two state-of-the-art methods described in Subsection 4.4. When analyzing our personalized calibration proposal in comparison to CP calibration, we observe that our proposal achieved better values

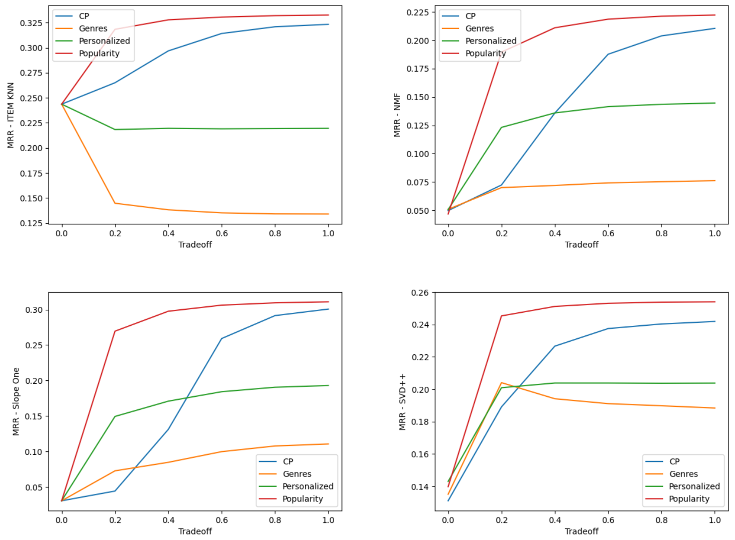

Fig. 6. MRR results over the selected recommenders and three types of calibration on the MovieLens 20M dataset.

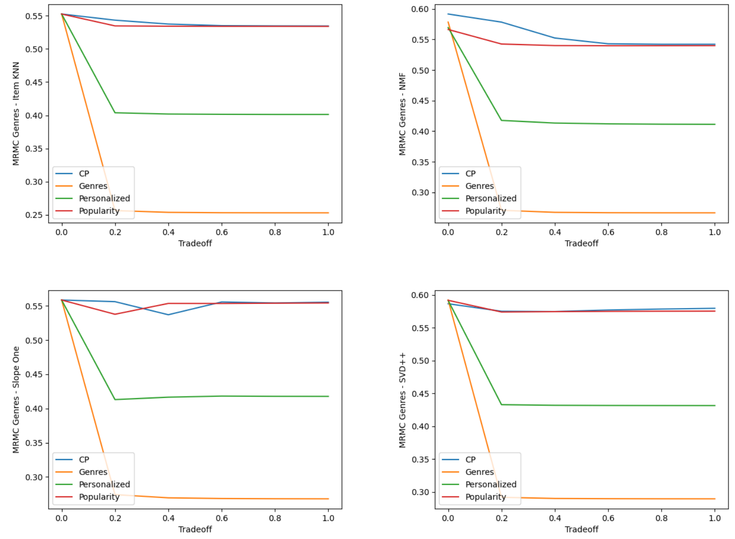

Fig. 7. MRMC of genres results over the selected recommenders and three types of calibration on the MovieLens 20M dataset.

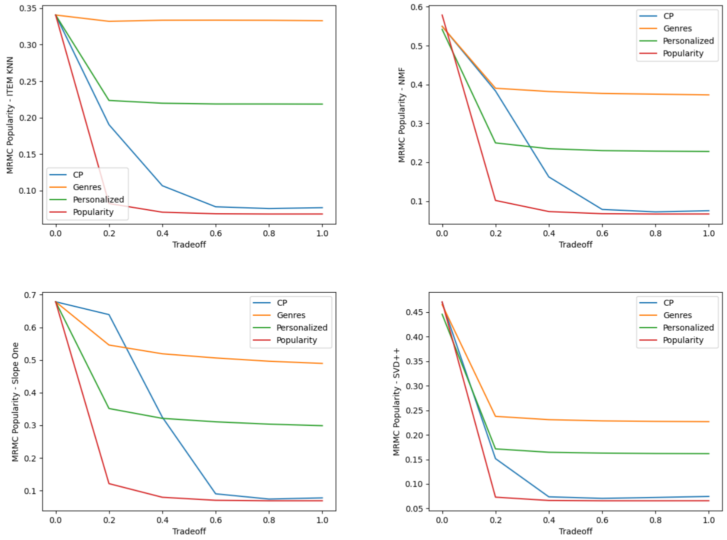

Fig. 8. MRMC of popularity results over the selected recommenders and three types of calibration on the MovieLens 20M dataset.

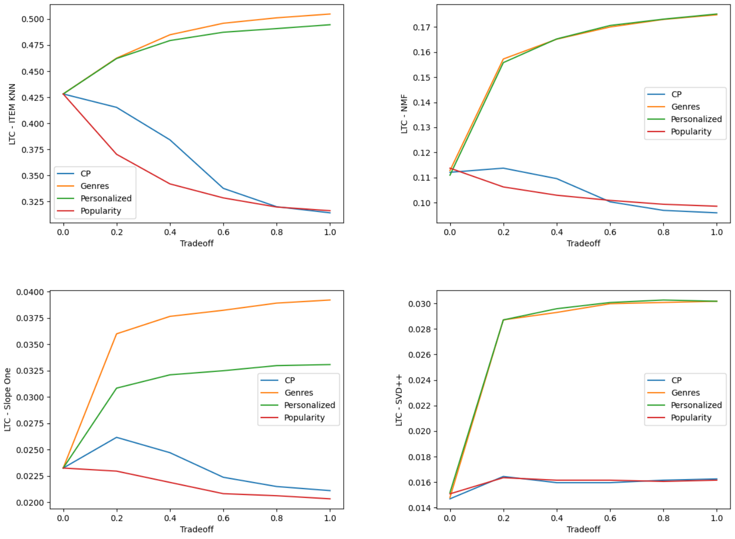

Fig. 9. LTC results over the selected recommenders and three types of calibration on the MovieLens 20M dataset.

Table 3. Comparison of our proposed calibration approaches against the baselines in the MovieLens 20M dataset. The ▲ symbol means a statistically significant improvement of the proposed approach in comparison to baselines, with a p-value < 0.05 using the Student's t-test; the ● symbol denotes no statistically significant gain or loss; and the ▼ symbol indicates that the baseline is statistically better than our proposal. Each pair of symbols is related to the CP and Genres baselines, respectively.

Algorithm	**LTC**	**MRMC Genres**	**MRMC Pop.**	**F1 Score**	**MRR**	**MAP**	ΔGAP_{BB}	ΔGAP_{N}	ΔGAP_{D}	**RMSE**
Item KNN + CP ($\lambda = 0.0$)	0.428	0.553	0.341	0.531	0.244	0.105	-0.666	0.501	-0.471	0.319
NMF + CP ($\lambda = 0.2$)	0.114	0.578	0.383	0.499	0.072	0.028	-0.859	-0.359	-0.698	0.392
SlopeOne + CP ($\lambda = 0.2$)	0.026	0.556	0.639	0.396	0.044	0.017	-0.939	-0.673	-0.844	0.485
SVD++ + CP ($\lambda = 0.2$)	0.016	0.575	0.152	0.565	0.189	0.084	-0.623	0.153	-0.366	0.246
Item KNN + Genres ($\lambda = 1.0$)	0.505	0.253	0.333	0.705	0.134	0.053	-0.797	0.207	-0.632	0.347
NMF + Genres ($\lambda = 1.0$)	0.175	0.267	0.373	0.676	0.076	0.03	-0.809	0.095	-0.647	0.345
SlopeOne + Genres ($\lambda = 1.0$)	0.039	0.268	0.49	0.602	0.111	0.046	-0.893	0.001	-0.57	0.353
SVD++ + Genres ($\lambda = 1.0$)	0.03	0.289	0.227	0.748	0.188	0.083	-0.667	0.762	-0.311	0.353
Item KNN + Personalized ($\lambda = 1.0$)	0.495 ▲ ▼	0.401	0.219	0.678 ▲ ▼	0.220 ▼ ▲	0.101 ▼ ▲	-0.797	0.274	-0.385	0.310 ▲ ▲
NMF + Personalized ($\lambda = 1.0$)	0.175 ▲ ●	0.411	0.228	0.668 ▲ ▼	0.145 ● ▲	0.063 ● ▲	-0.785	0.282	-0.372	0.302 ▲ ▲
SlopeOne + Personalized ($\lambda = 1.0$)	0.033 ▲ ▼	0.418	0.299	0.640 ▲ ▲	0.193 ● ▲	0.090 ● ▲	-0.893	0.374	-0.295	0.340 ▲ ▲
SVD++ + Personalized ($\lambda = 1.0$)	0.030 ▲ ▲	0.431	0.162	0.678 ▲ ▼	0.204 ▲ ▲	0.090 ▲ ▲	-0.627	0.287	-0.198	0.238 ▲ ▲
Item KNN + Popularity ($\lambda = 0.0$)	0.428 ▲ ▼	0.553	0.341	0.531 ▲ ▼	0.244 ▲ ▲	0.105 ▲ ▲	-0.666	0.501	-0.471	0.319 ● ▲
NMF + Popularity ($\lambda = 0.0$)	0.114 ▲ ▼	0.566	0.579	0.424 ▼ ▼	0.047 ▼ ▼	0.017 ▼ ▼	-0.905	-0.167	-0.801	0.407 ▼ ▼
SlopeOne + Popularity ($\lambda = 1.0$)	0.023 ▼ ▼	0.559	0.068	0.603 ▲ ▲	0.031 ▼ ▼	0.011 ▼ ▼	-0.996	0.460	-0.896	0.134 ● ▲
SVD++ + Popularity ($\lambda = 1.0$)	0.016 ▲ ▼	0.575	0.066	0.584 ▲ ▼	0.254 ▲ ▲	0.123 ▲ ▲	-0.063	0.300	0.040	0.102 ● ▲

for the LTC, F1, and RMSE metrics in all cases, indicating a more diverse and fairer recommendation list.

When compared to the genre-based calibration [22], our personalized calibration was superior for all recommenders in terms of accuracy, as shown by the MRR and MAP metrics, and also fairness, as shown by the RMSE metric.

Regarding the popularity-based calibration proposal, it demonstrated to be superior to CP in LTC, F1, MRR, and MAP for the majority of recommenders and fairer than genre-based calibration in terms of RMSE for most recommenders.

5.3 Yahoo Songs

Mean Reciprocal Rank. Figure 10 shows the comparison among the approaches in terms of MRR in the Yahoo Songs dataset. Similarly to the MovieLens 20M dataset, our popularity-based calibration approach obtained the best accuracy throughout the trade-off values.

Genre Mean Rank Miscalibration. Figure 11 shows the results of MRMC related to genres in the Yahoo Songs dataset. Like the other datasets, when $\lambda \geq 0.1$, all methods enhanced genre-related fairness. However, genre-based calibration outperformed the others, as it was initially tailored to ensure fairness based on genres. While our proposed method did not attain the highest level of MRMC, it yielded superior outcomes compared to both CP and our popularity-based approaches across all scenarios.

Popularity Mean Rank Miscalibration. Figure 12 shows the MRMC results related to popularity. Analogous to the observations made in the Yahoo Movies dataset, our proposed popularity calibration outperformed fairness associated with popularity across all recommenders, including the CP calibration method.

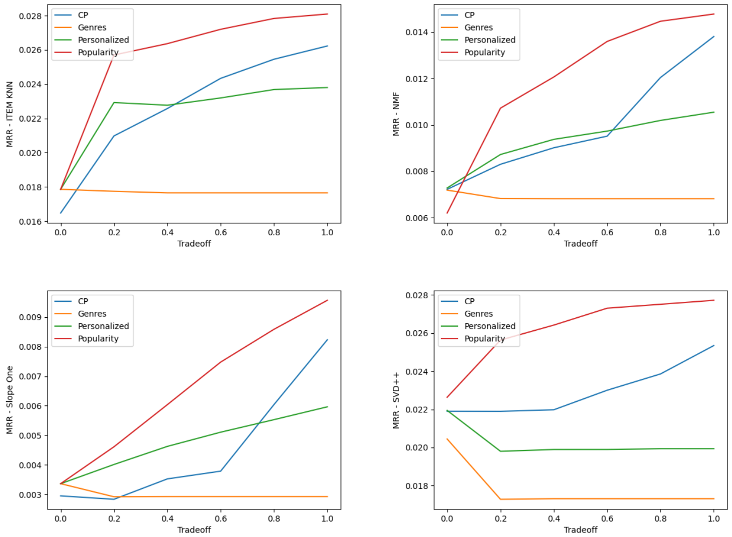

Fig. 10. MRR results over the selected recommenders and three types of calibration on the Yahoo Songs dataset.

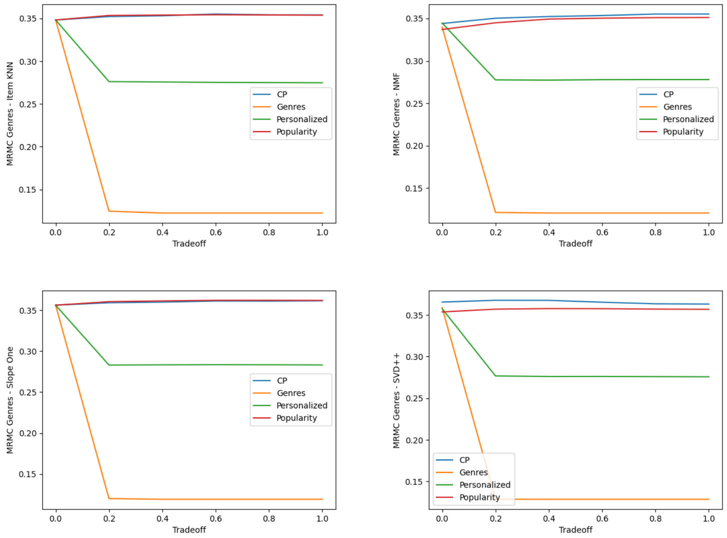

Fig. 11. MRMC of genres results over the selected recommenders and three types of calibration on the Yahoo Songs dataset.

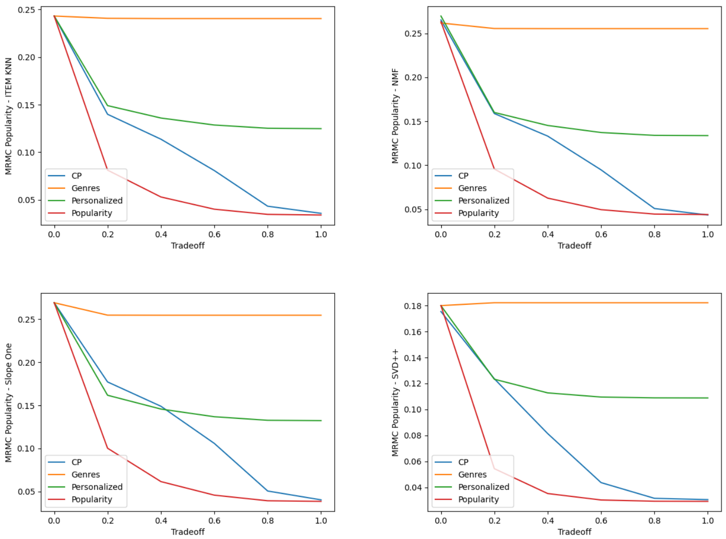

Fig. 12. MRMC of popularity results over the selected recommenders and three types of calibration on the Yahoo Songs dataset.

Long Tail Coverage. Figure 13 shows the results of LTC on the Yahoo Songs dataset. The CP calibration method obtained the lowest results, indicating a lack of diversity on the recommendation list. The proposed personalized-based calibration achieved the best results across the trade-off weights in the NMF and SlopeOne recommenders, and genre-based calibration was able to improve Item KNN and SVD++ regarding this metric.

Baselines Comparison. Table 4 compares our proposed personalized calibration method with the two state-of-the-art methods described in Subsection 4.4. It can be noted that our personalized calibration, compared to CP, yielded superior results for the LTC and F1 metrics, indicating more diverse and better calibration regarding genres and popularity. However, CP was able to provide better fairness than personalized-based calibration, as shown by the RMSE metric.

When compared to genre-based calibration, the personalized-based calibration was superior in terms of MRR and MAP but could not outperform this baseline in F1 and RMSE metrics.

In comparison with our popularity-based proposal, we achieved superior results over CP calibration for the LTC metric, indicating greater diversity. Furthermore, when combined with SlopeOne and SVD++, we obtained higher values for MAP, MRR, and MRMC Genres, demonstrating good accuracy and greater fairness in terms of genres. Finally, concerning genre-based calibration, we achieved higher values for the LTC, MRMC Pop., MAP, and MRR metrics, confirming good accuracy, diversity, and fairness in terms of popularity.

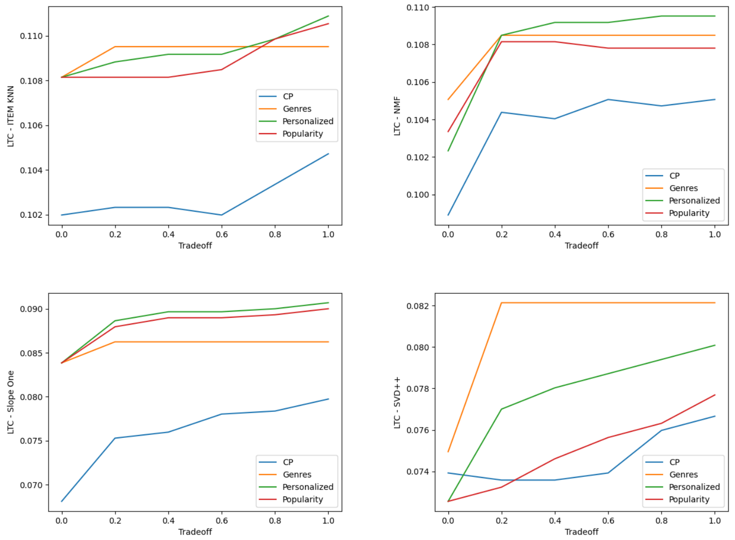

Fig. 13. LTC results over the selected recommenders and three types of calibration on the Yahoo Songs dataset.

6 Conclusion

In this paper, we proposed two calibration techniques, popularity-based and personalized-based, which use popularity and genre calibrations in a switch-based approach to provide fairer recommendations to users according to their interests. As an extension to [20], we assessed in this paper a deep analysis of our techniques using additional recommender models and a dataset from a different domain. We also evaluate the calibration approaches using a set of metrics that measure aspects such as accuracy, diversity, fairness, and miscalibration.

The experiments showed that the proposed techniques were able to improve the baselines in many aspects, although some challenges still need to be investigated, such as the trade-off between diversity and precision [14]. Despite this, we were able to demonstrate that our calibration approaches could be applied in a different domain, with improvements in diversity, accuracy, and fairness.

In future work, we plan to analyze the effect of our calibration with other recommendation models, particularly considering the aspects of precision and popularity bias. We will also conduct online experiments to verify the performance of the proposed calibration system with real users.

ACKNOWLEDGEMENTS

The authors would like to thank the financial support from FAPESP, process number 2022/07016-9.

Table 4. Comparison of our proposed calibration approaches against the baselines in the Yahoo Songs dataset. The ▲ symbol means a statistically significant improvement of the proposed approach in comparison to baselines, with a p-value < 0.05 using the Student's t-test; the ● symbol denotes no statistically significant gain or loss; and the ▼ symbol indicates that the baseline is statistically better than our proposal. Each pair of symbols is related to the CP and Genres baselines, respectively.

Algorithm	**LTC**	**MRMC Genres**	**MRMC Pop.**	**F1 Score**	**MRR**	**MAP**	ΔGAP_{BB}	ΔGAP_{N}	ΔGAP_{D}	**RMSE**
Item KNN + CP ($\lambda = 1.0$)	0.105	0.354	0.036	0.774	0.026	0.01	-0.357	-0.807	-0.675	0.370
NMF + CP ($\lambda = 1.0$)	0.105	0.355	0.044	0.771	0.014	0.005	-0.592	-0.811	-0.72	0.412
SlopeOne + CP ($\lambda = 1.0$)	0.08	0.362	0.04	0.767	0.008	0.003	-0.551	-0.814	-0.736	0.409
SVD++ + CP ($\lambda = 1.0$)	0.077	0.363	0.003	0.769	0.025	0.009	-0.167	-0.803	-0.635	0.346
Item KNN + Genres ($\lambda = 1.0$)	0.11	0.122	0.24	0.814	0.018	0.006	-0.823	-0.630	-0.735	0.424
NMF + Genres ($\lambda = 1.0$)	0.108	0.121	0.255	0.806	0.007	0.002	-0.851	-0.706	-0.787	0.453
SlopeOne + Genres ($\lambda = 1.0$)	0.086	0.119	0.255	0.807	0.003	0.001	-0.853	-0.696	-0.798	0.453
SVD++ + Genres ($\lambda = 1.0$)	0.082	0.129	0.182	0.845	0.017	0.006	-0.763	-0.604	-0.639	0.388
Item KNN + Personalized ($\lambda = 1.0$)	0.111 ▲ ●	0.275	0.125	0.793 ▲ ▼	0.024 ● ▲	0.009 ● ▲	-0.823	-0.807	-0.672	0.445 ▼ ▼
NMF + Personalized ($\lambda = 1.0$)	0.110 ▲ ●	0.278	0.134	0.788 ▲ ▼	0.011 ● ▲	0.004 ● ▲	-0.853	-0.807	-0.728	0.460 ▼ ▼
SlopeOne + Personalized ($\lambda = 1.0$)	0.091 ▲ ▲	0.283	0.132	0.785 ▲ ▼	0.006 ● ▲	0.002 ● ▲	-0.853	-0.814	-0.739	0.464 ▼ ▼
SVD++ + Personalized ($\lambda = 1.0$)	0.080 ▲ ▼	0.276	0.109	0.799 ▲ ▼	0.020 ▼ ▲	0.007 ▼ ▲	-0.738	-0.802	-0.608	0.415 ▼ ▼
Item KNN + Popularity ($\lambda = 1.0$)	0.111 ▲ ●	0.354	0.034	0.774 ▲ ▼	0.028 ▲ ▲	0.010 ▲ ▲	-0.357	-0.807	-0.573	0.351 ▲ ▲
NMF + Popularity ($\lambda = 0.4$)	0.108 ▲ ▲	0.349	0.063	0.768 ▼ ▼	0.012 ▼ ▲	0.004 ▼ ▲	-0.556	-0.793	-0.690	0.396 ▲ ▲
SlopeOne + Popularity ($\lambda = 1.0$)	0.090 ▲ ▲	0.362	0.038	0.767 ▲ ▼	0.010 ▲ ▲	0.004 ▲ ▲	-0.551	-0.814	-0.658	0.394 ▲ ▲
SVD++ + Popularity ($\lambda = 1.0$)	0.078 ● ▼	0.357	0.029	0.774 ▲ ▼	0.028 ▲ ▲	0.010 ▲ ▲	-0.320	-0.798	-0.519	0.334 ▲ ▲

References

1. Abdollahpouri, H., Mansoury, M., Burke, R. & Mobasher, B. The unfairness of popularity bias in recommendation. *ArXiv Preprint ArXiv:1907.13286*. (2019)
2. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B. & Malthouse, E. User-centered evaluation of popularity bias in recommender systems. *Proceedings Of The 29th ACM Conference On User Modeling, Adaptation And Personalization*. pp. 119-129 (2021)
3. Abdollahpouri, H., Burke, R. & Mobasher, B. Popularity-Aware Item Weighting for Long-Tail Recommendation. (arXiv,2018), https://arxiv.org/abs/1802.05382
4. Beutel, A., Chen, J., Doshi, T., Qian, H., Wei, L., Wu, Y., Heldt, L., Zhao, Z., Hong, L., Chi, E. & Others Fairness in recommendation ranking through pairwise comparisons. *Proceedings Of The 25th ACM SIGKDD International Conference On Knowledge Discovery Data Mining*. pp. 2212-2220 (2019)
5. Boratto, L., Fenu, G. & Marras, M. Connecting user and item perspectives in popularity debiasing for collaborative recommendation. *Information Processing Management*. **58**, 102387 (2021)
6. Borges, R. & Stefanidis, K. On Mitigating Popularity Bias in Recommendations via Variational Autoencoders. *Proceedings Of The 36th Annual ACM Symposium On Applied Computing*. pp. 1383-1389 (2021), https://doi.org/10.1145/3412841.3442123
7. Cha, S. Comprehensive Survey on Distance/Similarity Measures between Probability Density Functions. *International Journal Of Mathematical Models And Methods In Applied Sciences*. **1**, 300-307 (2007), http://www.gly.fsu.edu/ parker/geostats/Cha.pdf
8. Chen, J., Dong, H., Wang, X., Feng, F., Wang, M. & He, X. Bias and debias in recommender system: A survey and future directions. *ACM Transactions On Information Systems*. **41**, 1-39 (2023)
9. Chen, Z., Wu, J., Li, C., Chen, J., Xiao, R. & Zhao, B. Co-training disentangled domain adaptation network for leveraging popularity bias in recommenders. *Proceedings Of The 45th International ACM SIGIR Conference On Research And Development In Information Retrieval*. pp. 60-69 (2022)

10. Da Silva, D., Manzato, M. & Durão, F. Exploiting personalized calibration and metrics for fairness recommendation. *Expert Systems With Applications.* **181** pp. 115112 (2021), https://www.sciencedirect.com/science/article/pii/S0957417421005534
11. Geyik, S., Ambler, S. & Kenthapadi, K. Fairness-aware ranking in search recommendation systems with application to linkedin talent search. *Proceedings Of The 25th Acm Sigkdd International Conference On Knowledge Discovery Data Mining.* pp. 2221-2231 (2019)
12. Kaya, M. & Bridge, D. A comparison of calibrated and intent-aware recommendations. *Proceedings Of The 13th ACM Conference On Recommender Systems.* pp. 151-159 (2019)
13. Koren, Y. Factorization Meets the Neighborhood: A Multifaceted Collaborative Filtering Model. *Proceedings Of The 14th ACM SIGKDD International Conference On Knowledge Discovery And Data Mining.* pp. 426-434 (2008), https://doi.org/10.1145/1401890.1401944
14. Landin, A., Suárez-García, E. & Valcarce, D. When Diversity Met Accuracy: A Story of Recommender Systems. *Proceedings.* **2** (2018), https://www.mdpi.com/2504-3900/2/18/1178
15. Lesota, O., Melchiorre, A., Rekabsaz, N., Brandl, S., Kowald, D., Lex, E. & Schedl, M. Analyzing item popularity bias of music recommender systems: Are different genders equally affected?. *Fifteenth ACM Conference On Recommender Systems.* pp. 601-606 (2021)
16. Lin, A., Wang, J., Zhu, Z. & Caverlee, J. Quantifying and mitigating popularity bias in conversational recommender systems. *Proceedings Of The 31st ACM International Conference On Information Knowledge Management.* pp. 1238-1247 (2022)
17. Luo, X., Zhou, M., Xia, Y. & Zhu, Q. An Efficient Non-Negative Matrix-Factorization-Based Approach to Collaborative Filtering for Recommender Systems. *IEEE Transactions On Industrial Informatics.* **10**, 1273-1284 (2014)
18. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K. & Galstyan, A. A survey on bias and fairness in machine learning. *ACM Computing Surveys (CSUR).* **54**, 1-35 (2021)
19. Naghiaei, M., Rahmani, H. & Dehghan, M. The unfairness of popularity bias in book recommendation. *International Workshop On Algorithmic Bias In Search And Recommendation.* pp. 69-81 (2022)
20. Sacilotti, A.; Souza, R. & Manzato, M. Counteracting Popularity-Bias and Improving Diversity Through Calibrated Recommendations. *Proceedings of the 25th International Conference on Enterprise Information Systems - Volume 1.* ISBN 978-989-758-648-4, ISSN 2184-4992, pp. 709-720 (2023)
21. Seymen, S., Abdollahpouri, H. & Malthouse, E. A Constrained Optimization Approach for Calibrated Recommendations. *Fifteenth ACM Conference On Recommender Systems.* pp. 607-612 (2021)
22. Steck, H. Calibrated Recommendations. *Proceedings Of The 12th ACM Conference On Recommender Systems.* pp. 154-162 (2018), https://doi.org/10.1145/3240323.3240372
23. Hug, N. Surprise: A Python library for recommender systems. *Journal Of Open Source Software.* **5**, 2174 (2020), https://doi.org/10.21105/joss.02174
24. Verma, S., Gao, R. & Shah, C. Facets of fairness in search and recommendation. *International Workshop On Algorithmic Bias In Search And Recommendation.* pp. 1-11 (2020)

25. Wang, W., Feng, F., He, X., Wang, X. & Chua, T. Deconfounded recommendation for alleviating bias amplification. *Proceedings Of The 27th ACM SIGKDD Conference On Knowledge Discovery Data Mining*. pp. 1717-1725 (2021)
26. Wei, T., Feng, F., Chen, J., Wu, Z., Yi, J. & He, X. Model-Agnostic Counterfactual Reasoning for Eliminating Popularity Bias in Recommender System. *Proceedings Of The 27th ACM SIGKDD Conference On Knowledge Discovery Data Mining*. pp. 1791-1800 (2021), https://doi.org/10.1145/3447548.3467289
27. Winter, J. Using the Student's t-test with extremely small sample sizes. *Practical Assessment, Research Evaluation*. **18** (2013,8)
28. Yalcin, E. Blockbuster: A new perspective on popularity-bias in recommender systems. *2021 6th International Conference On Computer Science And Engineering (UBMK)*. pp. 107-112 (2021)
29. Yalcin, E. & Bilge, A. Investigating and counteracting popularity bias in group recommendations. *Information Processing Management*. **58**, 102608 (2021), https://www.sciencedirect.com/science/article/pii/S0306457321001047
30. Yalcin, E. & Bilge, A. Treating adverse effects of blockbuster bias on beyond-accuracy quality of personalized recommendations. *Engineering Science And Technology, An International Journal*. **33** pp. 101083 (2022)
31. Zhang, Y., Feng, F., He, X., Wei, T., Song, C., Ling, G. & Zhang, Y. Causal Intervention for Leveraging Popularity Bias in Recommendation. *Proceedings Of The 44th International ACM SIGIR Conference On Research And Development In Information Retrieval*. (2021,7), https://doi.org/10.1145%252F3404835.3462875
32. Zhu, Z., Wang, J. & Caverlee, J. Measuring and Mitigating Item Under-Recommendation Bias in Personalized Ranking Systems. *Proceedings Of The 43rd International ACM SIGIR Conference On Research And Development In Information Retrieval*. pp. 449-458 (2020), https://doi.org/10.1145/3397271.3401177