Mitigating Popularity Bias in Recommendation: Potential and Limits of Calibration Approaches

Anastasiia Klimashevskaia[1(✉)], Mehdi Elahi[1], Dietmar Jannach[2], Christoph Trattner[1], and Lars Skjærven[3]

[1] University of Bergen, Bergen, Norway
{anastasiia.klimashevskaia,mehdi.elahi,christoph.trattner}@uib.no
[2] University of Klagenfurt, Klagenfurt, Austria
dietmar.jannach@aau.at
[3] TV 2, Bergen, Norway
lars.skjerven@tv2.no

Abstract. While recommender systems are highly successful at helping users find relevant information online, they may also exhibit a certain undesired bias of mostly promoting only already popular items. Various approaches of quantifying and mitigating such biases were put forward in the literature. Most recently, *calibration* methods were proposed that aim to match the popularity of the recommended items with popularity preferences of individual users. In this paper, we show that while such methods are efficient in avoiding the recommendation of too popular items for some users, other techniques may be more effective in reducing the popularity bias on the platform level. Overall, our work highlights that in practice choices regarding metrics and algorithms have to be made with caution to ensure the desired effects.

Keywords: Recommender Systems · Bias · Multi-Metric Evaluation

1 Introduction and Background

The value of recommender systems, e.g., on e-commerce or media streaming sites—both for consumers and providers—is undisputed [12]. Yet, such systems may sometimes lead to the undesired effect that they mainly promote already popular items [9,10]. A strong *popularity bias* of underlying algorithms may lead to limited exposure of long-tail items through the recommendations and, ultimately, to limited discovery effects and missed sales or engagement opportunities [22].

Various algorithmic approaches were proposed in the literature to deal with such a bias [1,4,5]. Usually, this mitigation process involves handling a trade-off between predicted item relevance (accuracy) and item popularity. Other strategies are possible as well, including re-ranking of an accuracy-optimized list, introducing popularity aspects in the loss function or when sampling data during learning [6,13,21]. An important question in this context is how the popularity

L. Boratto et al. (Eds.): BIAS 2022, CCIS 1610, pp. 82–90, 2022.
https://doi.org/10.1007/978-3-031-09316-6_8

bias of an algorithm is quantified, i.e., which metrics are used. One commonly applied approach is to generate top-n recommendation lists for each user, and to then determine average popularity of the items in these lists [13].

Such an approach allows us to quantify the popularity bias on a *platform level*, i.e., across all users. Boratto et al. [3], for instance, studied different algorithms in the course recommendation domain, analyzing how certain techniques can amplify or mitigate biases within the system. Later, Kowald et al. [15] performed similar studies within the music domain.

In recent years, alternative approaches have been receiving more attention, which deal with biases on an *individual level*. The idea of such *calibration* approaches [14,19] is to create recommendation lists which match the individual user's past preference profile in terms of the distribution of certain item properties, e.g. item type, genre or popularity. Practically, the goal is therefore often to minimize the distance between two distributions, quantified, e.g., through the Kullback-Leibler divergence. In a recent work [2], Abdollahpouri et al. proposed and investigated the effectiveness of a particular user-specific approach named Calibrated Popularity (CP) for the mitigation of popularity biases.

Their proposed re-ranking technique aims to minimize the distance between two probability distributions, named UPD (User Popularity Deviation), as done earlier in [14,17,19]. As a main outcome of their experiments, the authors found that their method is not only effective in considering individual user tendencies, as expected by design, but may also help to improve existing metrics on the platform level.

The work in this paper is based on the needs of an industrial partner, the Norwegian broadcaster TV 2, who observed a significant popularity bias in their current recommendations [8]. One specific goal of the partner is thus to investigate the effectiveness of current popularity bias mitigation strategies. In this work, we present the results of such an analysis based both on a proprietary dataset from TV 2 and on a publicly available movie ratings dataset MovieLens. Going beyond existing works, we consider a selection of six "beyond-accuracy" metrics in our experiments to obtain a more fine-grained picture of the effects of three bias mitigation strategies. Moreover, like in [2], we consider mitigation re-ranking strategies of different types, including the recent CP approach.

In our analysis we could reproduce the findings from [2] regarding the effectiveness of CP with respect to the UPD criterion also for our additional dataset. However, it turns out that other methods are more effective when it comes to reducing the popularity bias on the *platform* level. From the perspective of a practitioner, the choice of the mitigation strategy should therefore be informed by the relative importance of the intended effects, i.e., if it is more important to match past consumer preference distributions or to increase the exposure of long-tail items. While the ultimate effects of the explored mitigation strategies on relevant Key Performance Indicators of TV 2 can only be determined through a field test, the offline analyses in this paper may serve as a basis for informing the choice of algorithms to be included in ongoing and future A/B tests at TV 2.

2 Research Methodology

To investigate the effectiveness of different bias mitigation strategies with respect to accuracy and popularity-related metrics, we ran extensive computational experiments. We describe the details of our experimental design in terms of considered algorithms, metrics, and datasets next.

2.1 Baseline Algorithms and Re-ranking Algorithms

Baseline Methods. In our study, we focus on re-ranking (post-processing) strategies for popularity bias mitigation, which take an accuracy-optimized list as a starting point. To generate this starting point, we use the ALS (Alternating Least Squares) method [20] for two main reasons. First, a version of this method is used by the industry partner as one of the techniques in their production systems. Second, ALS was also used as a baseline in [2]. We have systematically tuned the hyper-parameters of ALS for both datasets individually using grid search.

Besides ALS, we consider a simple and non-personalized popularity-based method for comparison in our study. This method, which we refer to as *Pop*, gives us an upper-bound in terms of some of the considered metrics.

Bias Mitigation Methods. In accordance with the objectives of the industry partner, we focus only on re-ranking (post-processing) methods in this study, leaving out the model-based ones. We have reused or re-implemented the following algorithms that were considered in [2]:

- **Calibrated Popularity [CP]**: This is the main method proposed in [2]. This calibration-based algorithm[1] personalizes the recommendations in terms of popularity of the recommended items, considering the previous preference history of every user separately. The algorithm differentiates between head, middle and (long) tail items.
- **Personalized Long Tail Promotion [XQ]**: This approach, originally proposed in [1], is based on the xQuAD algorithm [18] from IR. XQ aims to balance the proportion of head and tail items in recommendation lists by leveraging the user propensity towards popular items. Only two categories of items are distinguished in the method: head and tail items.
- **FA*IR [FS]**: This method, proposed in [24], gives "protected items" from the candidate list more exposure. In this particular case the protected group is represented by the tail items.

Definition of Item Popularity Groups. A common practice in the literature—and for the considered methods—is to split the items into different groups according to their level of popularity. Besides distinguishing between head and tail items, some works further split the tail items into the sub-groups, i.e., *middle* and

[1] Similar ideas were proposed earlier in [14] and [17], and later independently popularized under the term *calibration* in [19].

distant tail items [2]. In this work, we follow this latter approach, which allows us to focus on specific subgroups of items when conducting our analyses. For that, we first sort all items in descending order of their popularity, where we use the number of interactions per item in the data as a popularity indicator. Then we compute the sum of the corresponding (normalized) item popularity scores as *total_pop*. To create the set of *head* items, we add items from the top of the popularity sorted list until the sum of popularity scores in *head* reaches 20% of *total_pop*. Then, items from the end of the popularity-sorted list are added to the *tail* set until the set of tail items reaches 20% of *total_pop*. The remaining items then form the *middle* set of items.

2.2 Metrics

In the research literature, different metrics for quantifying popularity biases have been proposed. Since one goal of our study is to investigate the effects of bias mitigation strategies in a comprehensive way, we consider the following set of metrics:

- **User Popularity Deviation (UPD).** This metric was introduced in [2] as the average popularity deviation across different user groups G:

$$UPD = \frac{\sum_{g \in G} UPD(g)}{|G|}$$

with

$$UPD(g) = \frac{\sum_{u \in g} JSD(P(u), Q(u))}{|g|}$$

calculated for every user group defined in the algorithm. $JSD(P(u), Q(u))$ is the Jensen-Shannon divergence [16], which measures the distance between two probability distributions, wigh $P(u)$ being the popularity distribution of items in the user u profile and $Q(u)$ the popularity distribution in the recommendation list for the user u. See also [2] for more details and illustrative examples.
UPD indicates how well the re-ranking algorithm adjusts the recommendation to the user interest history, matching the distribution of head and tail items. UPD is the optimization goal of the CP method. Like earlier works, who used the Earth Mover's Distance [17] or the Kullback-Leibler divergence, UPD measures the difference between distributions, and lower values therefore mean a better personalization. Increasing UPD, however, does not necessarily lead to a much *lower* popularity on the platform level. Lovers of "blockbuster" movies, for example, would by design still receive many highly-popular movies as recommendations.
- **Average Recommendation Popularity (ARP).** This commonly used metric simply returns the average popularity of the items in the top-n recommendation lists produced by an algorithm for all users. This metric was defined in [23] as follows:

$$ARP = \frac{1}{|U|} \sum_{u \in U} \frac{\sum_{i \in L_u} \phi(i)}{|L_u|}$$

where $\phi(i)$ is the popularity of an item i, i.e., the number of ratings or interactions that are observed for it in the training set, and $L(u)$ is the list of items recommended to user u. This averaging metric should however be interpreted with care, since recommending only a few very unpopular items to everyone can lead to relatively low ARP values.

– **Average Percentage of Long Tail Items (APLT), Average Coverage of Long Tail items (ACLT).** In [1], these metrics are defined as follows:

$$APLT = \frac{1}{|U_t|} \sum_{u \in U_t} \frac{|\{i, i \in (L_u \cap \Gamma)\}|}{|L_u|}$$

$$ACLT = \frac{1}{|U_t|} \sum_{u \in U} \sum_{i \in L_u} 1(i \in \Gamma)$$

where Γ is the set of tail items. These metrics quantify the effect of the re-ranking with respect to long tail items. The first metric measures the average percentage of tail items in user recommendation lists, while the second one indicates the exposure of the tail items in the entire recommendation.

– **Aggregate Diversity and Gini Index.** In addition to UPD, Abdollahpouri et al. [2] report Aggregate Diversity and the Gini Index of the UPD-optimized recommendations, defining them as follows:

$$AggDiv = \frac{\cup_{u \in U} L_u}{|I|}$$

$$Gini(L) = 1 - \frac{1}{|I| - 1} \sum_{k=1}^{|I|} (2k - |I| - 1) p(i_k|L)$$

where L is the combined list of all the recommendations for all the users in U, and where $p(i|L)$ is the occurrence ratio of item i in L.

Aggregate Diversity informs about how many different items appear in recommendation lists of users, which is thus a form of *Item Coverage*. The Gini Index, in contrast quantifies how uneven the distribution of recommended items is. Higher values mean higher concentration.

Note that Item Coverage and the Gini index are not necessarily tied to popularity aspects. To obtain high Item Coverage, it is sufficient that many items appear at least once in a recommendation list. In terms of the Gini index, an algorithm that only recommends the most unpopular items to everyone would lead to high concentration, but not to a popularity bias. Realistically, however, we expect a higher concentration of short-head items for typical collaborative filtering algorithms. Usually, popularity metrics and measures like the Gini index are also often highly correlated with *novelty* metrics, as these are commonly based on popularity considerations, see [7].

Generally, we iterate that these metrics do not necessarily correlate, i.e., higher aggregate diversity may not necessarily mean a lower level of platform-wide popularity bias, as expressed, for example through the ARP measure. In terms of accuracy measures, we report *Precision* as done in [2] as well, which is also the target of our hyper-parameter optimization process.

2.3 Datasets

We have evaluated the different bias mitigation strategies on two datasets. First, we used a proprietary dataset provided by our industry partner TV 2. Second, like [2], we used a MovieLens dataset [11] to ensure reproducibility on publicly available data.[2] The dataset provided by TV 2 originally consisted of logged movie interaction data on the streaming of the provider. The recorded interactions, e.g., viewing times, were transformed into implicit feedback signals and a user-item interaction matrix by our industry partner. Given this dataset, we performed the same pre-processing steps as described in [2] including, for example, data filtering. The resulting dataset contains about 518K interactions by 9408 users on 1795 items. Since some of the examined algorithms, in particular CP, are computationally demanding, we have resorted to randomly sampling 1000 users for re-ranking and metric calculations. As done in [2] as well, we organized the datasets into an 80% training split and use the remaining 20% for testing.

3 Results

Table 1 shows the results of our evaluation. In terms of *accuracy*, we observe that all re-ranking methods, as expected, led to a small to modest decrease in Precision (about 2.0% to 4% for TV 2 and about 2.5% to 11% for MovieLens).

In terms of *UPD*, the CP method performs much better than the other techniques. Again, this is expected as CP directly aims to optimize for this *user-individual* metric. Looking at *platform-wide* metrics (ARP, APLT, ACLT), however, it turns out XQ leads to the strongest effects in popularity-bias reduction. Compared to the baseline (ALS), the average popularity of the recommended items (ARP), for example, goes down by at least 30%. The effects of the CP method on the ARP (and on the other platform-wide metrics) are, in contrast, much smaller, e.g., about 10% on the TV 2 dataset.[3]

Thus, when the goal is to mitigate platform-wide popularity effects, methods like XQ appear to be a better choice than user-centered calibration effects. In the reported experiments, XQ leads to a slightly stronger accuracy decrease than CP.

[2] Differently from [2], we used the MovieLens dataset with about 100k ratings by 943 users on 1612 items of in our experiments.

[3] Interestingly, in [2], CP was favorable over XQ also on the ARP measure. We could not reproduce this finding for both datasets. Unfortunately, the authors of [2] could not reproduce the code of the CP method. The observed discrepancy might therefore be both related to dataset characteristics and differences in the implementation.

Table 1. Evaluation results. Arrows indicate whether lower or higher values are better.

Dataset	Algorithm	Metrics						
		Accuracy	Calibration	Long Tail Exposure			Equal Exposure	
		Prec ↑	*UPD* ↓	*ARP* ↓	*APLT* ↑	*ACLT* ↑	*Agg-Div* ↑	*Gini* ↓
TV 2	**Pop**	0.301	0.644	0.301	0.000	0.000	0.006	0.994
	Base (ALS)	**0.875**	0.286	0.143	0.639	0.292	0.321	0.874
	XQ	0.818	0.358	**0.100**	**0.956**	**0.364**	0.343	0.850
	FS	0.857	0.249	0.126	0.772	0.299	0.328	0.856
	CP	0.837	**0.123**	0.130	0.672	0.314	**0.392**	**0.844**
ML	**Pop**	0.381	0.629	0.381	0.000	0.000	0.007	0.993
	Base (ALS)	**0.738**	0.261	0.243	0.634	0.391	0.425	0.851
	XQ	0.656	0.378	**0.167**	**0.989**	**0.442**	0.412	0.848
	FS	0.697	0.237	0.202	0.839	0.397	0.432	0.835
	CP	0.720	**0.101**	0.223	0.676	0.408	**0.477**	**0.821**

Depending on the application, XQ can however also be further tuned to focus more on accuracy. Such a tuning was however not the focus of our work.

Looking at the results for the remaining metrics, we find that all the re-ranking methods have at least a slight positive impact on Gini Index, with CP having the strongest effect. For Aggregate Diversity (Agg-Div) the values are mixed and depend on the dataset. Again, CP has the strongest effect across the methods. Remember, however, that Agg-Div and the Gini Index cannot inform us directly if popularity bias issues are successfully reduced.

4 Summary and Future Work

Our work highlights that calibrated recommendations—while being effective for matching the recommendations with past user tendencies—may not be the best possible choice when the goal is to mitigate platform-wide popularity bias effects. Thus, in practice, the choice of the bias mitigation algorithm and the popularity metrics have to be done with care and be dependent on the desired effects. If, for example, the general goal of the platform is to give the tail items more exposure, then the provider may consider XQ method, which performs best in terms of long tail exposure metrics in our study. However, this method should tuned with care, because in our experiments it led to the largest drop in accuracy compared to the other methods. If, on the other hand, the most important feature for the provider is to adjust the popularity distribution of the recommended items to the user's preference, then CP and similar approaches [14,19] are preferable. Yet, such approaches may not necessarily lead to a strong reduction of platform-wide popularity biases. In an extreme case, where all users are blockbuster lovers,

mainly the popular items would receive attention after calibration. Finally, if a provider is uncertain which of the aspects are more important to address, some middle-ground approach, e.g., a hybrid of the evaluated methods, may be adopted to balance the goals of reducing popularity biases at the platform level while considering individual past user popularity tendencies.

Overall, while the experiments reported in this paper lead to some interesting insights, some limitations remain, which we plan to address in our future work. First, we have so far only made experiments in one particular domain, that of movie recommendations. Second, also due to the requirements of the industry partners, only post-processing techniques—as opposed to *model-based* techniques—were investigated so far. Third, it could be intriguing as well to investigate different approaches to head-tail split of the items in the catalogue and how that might affect the outcome of the debiasing techniques application.

In our future work, we will therefore explore different domains and algorithms and furthermore design approaches that are able balance all three mentioned aspects: accuracy, individual-level and platform-wide effects. Moreover, we plan to evaluate the described methods in A/B tests with our industry partner.

Acknowledgement. This work was supported by industry partners and the Research Council of Norway with funding to MediaFutures: Research Centre for Responsible Media Technology and Innovation, through The Centres for Research-based Innovation scheme, project number 309339.

References

1. Abdollahpouri, H., Burke, R., Mobasher, B.: Managing popularity bias in recommender systems with personalized re-ranking. In: FLAIRS 2019, pp. 413–418 (2019)
2. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B., Malthouse, E.: User-centered evaluation of popularity bias in recommender systems. In: ACM UMAP 2021, pp. 119–129 (2021)
3. Boratto, L., Fenu, G., Marras, M.: The effect of algorithmic bias on recommender systems for massive open online courses. In: European Conference on Information Retrieval, pp. 457–472 (2019)
4. Boratto, L., Fenu, G., Marras, M.: Combining mitigation treatments against biases in personalized rankings: use case on item popularity. In: IIR 2021 (2021)
5. Boratto, L., Fenu, G., Marras, M.: Connecting user and item perspectives in popularity debiasing for collaborative recommendation. IP&M **58**(1), 102387 (2021)
6. Borges, R., Stefanidis, K.: On mitigating popularity bias in recommendations via variational autoencoders. In: ACM/SIGAPP SAC 2021, pp. 1383–1389 (2021)
7. Castells, P., Hurley, N.J., Vargas, S.: Novelty and diversity in recommender systems. In: Ricci, F., Rokach, L., Shapira, B. (eds.) Recommender Systems Handbook, pp. 881–918. Springer, New York (2015)
8. Elahi, M., Jannach, D., Skjærven, L., et al.: Towards responsible media recommendation. AI and Ethics (2021)
9. Elahi, M., Kholgh, D.K., Kiarostami, M.S., Saghari, S., Rad, S.P., Tkalcic, M.: Investigating the impact of recommender systems on user-based and item-based popularity bias. Inf. Process. Manage. **58**, 102655 (2021)

10. Fleder, D., Hosanagar, K.: Blockbuster culture's next rise or fall: the impact of recommender systems on sales diversity. Manage. Sci. **55**, 697–712, 102655 (2009)
11. Harper, F.M., Konstan, J.A.: The MovieLens datasets: history and context. ACM TIIS **5**(4), 1–19, 102655 (2015)
12. Jannach, D., Jugovac, M.: Measuring the business value of recommender systems. ACM Trans. Manage. Inf. Syst. **10**(4) (2019)
13. Jannach, D., Lerche, L., Kamehkhosh, I., Jugovac, M.: What recommenders recommend: an analysis of recommendation biases and possible countermeasures. User Model. User-Adap. Inter. **25**(5), 427–491 (2015). https://doi.org/10.1007/s11257-015-9165-3
14. Jugovac, M., Jannach, D., Lerche, L.: Efficient optimization of multiple recommendation quality factors according to individual user tendencies. Expert Syst. Appl. **81**, 321–331 (2017)
15. Kowald, D., Schedl, M., Lex, E.: The unfairness of popularity bias in music recommendation: a reproducibility study. In: European Conference on Information Retrieval, pp. 35–42 (2020)
16. Lin, J.: Divergence measures based on the Shannon entropy. IEEE Trans. Inf. Theory **37**(1), 145–151 (1991)
17. Oh, J., Park, S., Yu, H., Song, M., Park, S.T.: Novel recommendation based on personal popularity tendency. In: ICDM 2011, pp. 507–516 (2011)
18. Santos, R.L., Macdonald, C., Ounis, I.: Exploiting query reformulations for web search result diversification. In: WWW 2010, pp. 881–890 (2010)
19. Steck, H.: Calibrated recommendations. In: ACM RecSys 2018, pp. 154–162 (2018)
20. Takács, G., Tikk, D.: Alternating least squares for personalized ranking. In: ACM RecSys 2012, pp. 83–90 (2012)
21. Trattner, C., Elsweiler, D.: Investigating the healthiness of internet-sourced recipes: implications for meal planning and recommender systems. In: WWW 2017, pp. 489–498 (2017)
22. Trattner, C., et al.: Responsible media technology and AI: challenges and research directions. AI and Ethics, pp. 1–10 (2021)
23. Yin, H., Cui, B., Li, J., Yao, J., Chen, C.: Challenging the long tail recommendation. Proc. VLDB Endow. **5**(9), 896–907 (2012)
24. Zehlike, M., Bonchi, F., Castillo, C., Hajian, S., Megahed, M., Baeza-Yates, R.: FA*IR: a Fair Top-k ranking algorithm. In: CIKM 2017, pp. 1569–1578 (2017)