Fine-tuning for Inference-efficient Calibrated Recommendations

Oleg Lesota
Johannes Kepler University Linz
Linz, Austria
oleg.lesota@jku.at

Adrian Bajko
Johannes Kepler University Linz
Linz, Austria
bajko.adrian@gmail.com

Max Walder
Johannes Kepler University Linz
Linz, Austria
max@gstat.eu

Matthias Wenzel
Johannes Kepler University Linz
Linz, Austria
matthias.wenzel@rocketmail.com

Antonela Tommasel
ISISTAN
CONICET-UNCPBA
Tandil, Argentina
Johannes Kepler University Linz
Linz, Austria
antonela.tommasel@isistan.unicen.edu.ar

Markus Schedl
Johannes Kepler University Linz
Linz, Austria
markus.schedl@jku.at

Abstract

Calibration is the degree to which a recommender system is able to match the distribution of a certain item attribute among the items consumed by a user with their respective recommendations. Recent work suggests that many recommenders tend to provide miscalibrated recommendations. Furthermore, most approaches aimed at improving calibration adopt the post-processing paradigm, making them computationally costly at the inference time. This work proposes `CaliTune`, a fine-tuning approach applied to collaborative filtering based recommenders to allow them generate better calibrated recommendations without relying on costly post-processing. We compare `CaliTune` to an established post-processing approach on two backbone models and datasets from movie and music domains, focusing on popularity calibration. Our results suggest that `CaliTune` can offer a competitive accuracy–calibration trade-off in several settings, particularly when the backbone model exhibits high miscalibration and accuracy remains important, making it a promising inference-efficient alternative in such cases.

CCS Concepts

• **Information systems** → **Recommender systems**; **Personalization**; • **Social and professional topics** → *User characteristics.*

Keywords

Calibration, In-processing, Fine-tuning

ACM Reference Format:
Oleg Lesota, Adrian Bajko, Max Walder, Matthias Wenzel, Antonela Tommasel, and Markus Schedl. 2025. Fine-tuning for Inference-efficient Calibrated Recommendations. In *Proceedings of the Nineteenth ACM Conference on Recommender Systems (RecSys '25), September 22–26, 2025, Prague, Czech Republic.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3705328.3759319

RecSys '25, Prague, Czech Republic

ACM ISBN 979-8-4007-1364-4/25/09
https://doi.org/10.1145/3705328.3759319

1 Introduction

A recommender is considered calibrated with respect to an attribute (e.g., genre, release decade, or item popularity) if the distribution of the attribute in every user's recommendations matches its distribution in their consumption history. Certain aspects of calibration are being studied broadly, e.g., possible causes and temporal dynamics of miscalibration [18, 19], or alternative formulations such as rank-aware calibration [10]. Other studies focus on specific domains and attributes, for example, genre calibration in movies [25], country representation and calibration in music [15], and a growing body of research on popularity calibration [2, 3, 11, 12, 22], often studied alongside popularity bias [13]. The studies show that accuracy-optimized recommenders often provide miscalibrated recommendations [3, 14, 16].

Most calibration methods in the literature rely on post-processing, an additional step in the recommendation pipeline, where recommendations provided by a backbone recommender are re-ranked according to a certain rule. While effective, post-processing generally has to be performed individually for each user at the time of recommendation, thus introducing additional computational overhead at the inference time. Some hybrid approaches involving both in- and post-processing steps [8] or extending the backbone architecture [5] have been proposed for sequential recommendation. Nonetheless, for the classical top-K recommendation task, post-processing remains the dominant approach [3, 4, 22–25]. In particular, *Calibrated Popularity* [3] has proven effective for calibrating recommendations to users' popularity preferences [11, 12].

In this work, we explore *whether calibration can be improved solely through fine-tuning, without introducing inference-time overhead or altering the recommender's architecture.* As calibration is a slate-level property, improving it would require list-wise optimization. Yet, many established recommenders are trained with point- and pair-wise objectives, making the direct incorporation of list-wise calibration goals into their training process challenging. From this perspective, fine-tuning provides a flexible middle ground. We introduce `CaliTune`, a fine-tuning method designed to improve calibration of collaborative filtering based recommenders. The method combines large-scale dynamic sampling of candidate items per user, a heuristic list-wise loss, and a utility-aimed objective. In our

experiments, we apply CaliTune to two standard recommenders: Bayesian Personalized Ranking [20] (BPR), and Deep Matrix Factorization [26] (DMF). We evaluate CaliTune in two domains, music and movies, with a focus on popularity calibration, assessing its ability to align recommendations with users' individual preferences over item popularity. We address the following research questions:

• **RQ1: (Effectiveness)** Can CaliTune match the accuracy-calibration trade-off achieved by a SotA post-processing method?

• **RQ2: (Sensitivity)** To what extent does the accuracy–calibration trade-off of CaliTune vary across different fine-tuning configurations and user groups?

To the best of our knowledge, this is the first work to address calibration purely through fine-tuning, without modifying the backbone model or the recommendation pipeline. While here we experiment with calibrating popularity distributions, our broad goal is to *spark the discussion around fine-tuning as a general strategy for calibration, beyond post-processing or pipeline changes*. The proposed approach is not bound to popularity and can be applied to other calibration dimensions (e.g., genre), a larger number of attribute categories, other application domains and other model architectures.

2 Fine-tuning for calibration

In this paper, we introduce CaliTune, a fine-tuning approach for improving calibration of recommender systems, designed with deployment efficiency in mind. Commonly, calibration is implemented in the post-processing step as re-ranking, e.g. *Calibrated Popularity* (CP) [3]. However post-processing introduces additional computational overhead at the inference time, a major limitation for latency-sensitive applications or low-resource deployments. While post-processing approaches only affect the model's output, CaliTune integrates calibration as a fine-tuning step, allowing flexible updates to either the whole model or specific parts. This training-time integration eliminates the need for post-hoc adjustments, enabling fast and scalable deployment.

Our proposed fine-tuning procedure adjusts a pre-trained recommender to improve item-attribute calibration of its output. CaliTune uses the same train-validation-test split used for training the backbone model, ensuring a consistent evaluation framework. At each fine-tuning epoch, CaliTune dynamically generates top-m_{cand} candidate items for each user. The selection of these candidate items is based on the recommendation scores generated by the current state of the fine-tuned model. Here, $m_{cand} >= n$, where n represents the length of the recommendation list used for the final evaluation. We denote the full item collection $\mathcal{I}$, where each item belongs to one of cc categories (e.g., popularity categories $c \in \{HighPop, MidPop, LowPop\}$, $cc = 3$). This is encoded in matrix $C \in \{0,1\}^{|\mathcal{I}| \times cc}$. For each user u, we define $t_u \in \{0,1\}^{|\mathcal{I}| \times 1}$ (vector encoding the consumption history of u), $s_u \in \{\mathbb{Q}\}^{m_{cand} \times 1}$ (vector of recommendation scores for u's candidate items of the epoch), and matrix $C_u \in \{0,1\}^{m_{cand} \times cc}$ of one-hot encoded category assignments, for each candidate item of u.

The fine-tuning objective is to minimize the user-specific loss, balancing accuracy and calibration: $\mathcal{L}_u = (1-\lambda) \cdot \mathcal{LA}_u + \lambda \cdot \mathcal{LC}_u$. We define a heuristic calibration loss term $\mathcal{LC}_u$ (Eq. 1) as the Kullback-Leibler (KL) divergence between the target distribution H_u and the score-weighted attribute distribution R^*_u over the m_{cand} candidate items recommended to u. We compute H_u, the attribute distribution over the user's history, as in Eq. 2. The score-weighted distribution of the attribute over the candidates R^*_u is computed as in Eq. 3, where $\tilde{s_u}$ is a min-max normalized version of s_u. The contribution of each candidate item to its corresponding attribute category c in R^*_u is weighted by its recommendation score. This formulation aims to encourage the model to surface items from categories that are underrepresented in the recommendations (in relation to the consumption history) of the user u by increasing their recommendation scores relatively to items from other categories.

$$\mathcal{LC}_u = \sum_c H_u(c) \cdot \log \frac{H_u(c)}{R^*_u(c)} \quad (1) \qquad H_u = \frac{C^T t_u}{\sum_c (C^T t_u)_c} \quad (2) \qquad R^*_u = \frac{C_u^T \tilde{s_u}}{\sum_c (C_u^T \tilde{s_u})_c} \quad (3)$$

We define the accuracy-aimed loss term following [27], using the KL divergence between the softmax-normalized versions of s_u and tc_u (denoted $\hat{s_u}$ and $\hat{tc_u}$ respectively), where $tc_u \in \{0,1\}^{m_{cand} \times 1}$ is a vector encoding which of the candidate items are present in the user's consumption history (training positives). The accuracy loss term $\mathcal{LA}_u = KL(\hat{tc_u}, \hat{s_u})$, encourages the model to assign higher scores to the ground truth items relative to all other candidates. During fine-tuning, we compute the total loss for a batch as the sum of the user-level losses across all users in the batch.

The CaliTune hyper-parameters include the number of candidate items sampled at each epoch (m_{cand}), the weight given to the calibration objective (λ), and the scope of fine-tuning (whether all model parameters or, e.g., only the user embeddings are updated).

3 Experimental setup

Aiming to assess the feasibility of fine-tuning established recommenders for calibration, we conduct a series of exploratory experiments[1] featuring two backbone recommenders, two datasets from different domains, a SotA baseline and a user-group level analysis, targeting popularity as the calibration attribute[2]. The calibration task, therefore, is to adjust every user's individual recommendations such that items of every degree of popularity are represented there in the same proportions as in the user's consumption history, as formulated in previous work [3].

Item popularity. Following previous work [3, 11, 14], we define item popularity based on the total number of unique users who interacted with the item in the dataset. Each item is assigned to one of three popularity categories $c \in \{HighPop, MidPop, LowPop\}$. Items are ranked by popularity, and those at the top of the list cumulatively accounting for 20% of all interactions are labeled as $HighPop$. Similarly, the least popular items comprising 20% of interactions are labeled as $LowPop$. The remaining items fall into the $MidPop$ category. This categorization allows us to represent a user's popularity distribution, whether based on their consumption history (H_u) or recommendations (R_u), as a discrete three-bin. Each bin reflects the proportion of items from the respective category.

Baseline and evaluation. Following previous work [3, 14, 15], we quantify miscalibration at the user level using Jensen-Shannon

[1] Fostering reproducibility we share the experimental setup and results, including our implementations of CaliTune and CP: https://github.com/hcai-mms/CaliTune

[2] Popularity was selected as a clear, interpretable attribute with real-world relevance and a measurable trade-off with accuracy.

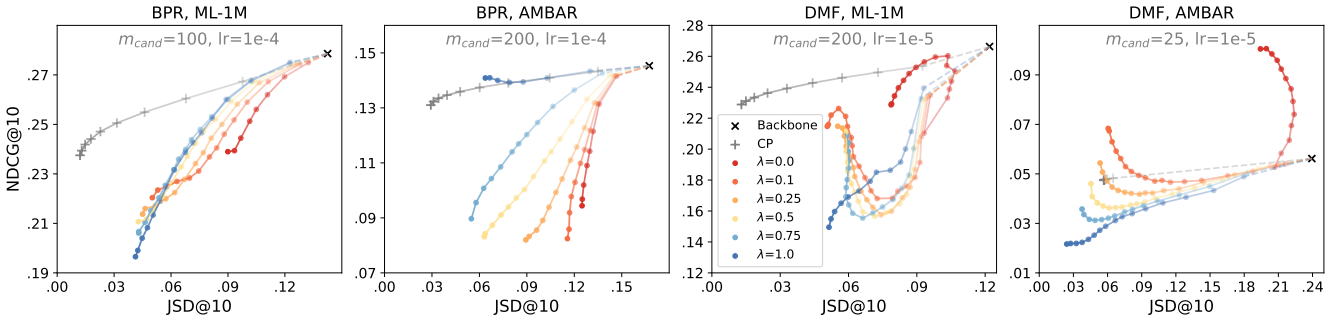

Figure 1: Accuracy-calibration trade-offs of `CaliTune` for different backbones, datasets, and values of the hyper-parameter λ. `CaliTune` is most effective on the AMBAR dataset, optimal settings vary between backbone-dataset combinations. Paler markers (situated closer to the backbone) correspond to earlier fine-tuning epochs for `CaliTune` and lower values of λ_{CP} for CP.

Table 1: Dataset statistics for AMBAR and ML-1M

	Users	Items	Interactions	Sparsity	*HighPop*	*MidPop*	*LowPop*
ML-1M	6,040	3,706	1,000,209	95.53%	113	1,069	2,524
AMBAR	30,618	69,372	1,739,596	99.92%	814	23,689	44,869

Divergence (*JSD*) between the popularity distributions in the user's top-10 recommendations and consumption history (Eq. 4). We choose it over *KL*, used in the fine-tuning loss for its directional optimization properties, for interpretability. In our formulation with $\log_2$ base *JSD* is bound to the $[0, 1]$ interval [17] and is symmetric. The *JSD* value of 0 indicates perfect alignment between recommended and historically consumed category distributions. We assess recommendation accuracy using *nDCG*@10.

We compare `CaliTune` to *Calibrated Popularity* (CP), a SotA user-centered post-processing method for popularity calibration [3]. It relies on greedy submodular optimization algorithm to re-rank the top-m candidate recommendations generated by a backbone recommender model, producing a final recommendation list of length n ($n << m$) for each user. The procedure optimizes an objective balancing the relevance and calibration goals with a hyper-parameter λ_{CP}. The computational cost of CP grows polynomially with the recommendation list length n, making this method relatively expensive at the inference time. In contrast, `CaliTune` integrates calibration into the training process, producing calibrated recommendations at the inference time without additional computational overhead.

$$JSD(H_u, R_u) = \frac{1}{2}\sum_c H_u(c) \cdot \log_2 \frac{2H_u(c)}{H_u(c)+R_u(c)} + \frac{1}{2}\sum_c R_u(c) \cdot \log_2 \frac{2R_u(c)}{H_u(c)+R_u(c)} \tag{4}$$

Datasets. We conduct our experiments on two datasets from different domains: MovieLens 1M (ML-1M, movies) [7], containing 1M interactions between 6K users and 4K items, and AMBAR (music) [6], which, after removing duplicate interactions and enforcing 5-core filtering, comprises 1.7M interactions between 31K users and 69K items. See Table 1 for detailed statistics.

Backbone models. We compare `CaliTune` to the post-proccesing baseline CP, applying both to the same backbone recommenders: `BPR` [21] and `DMF` [26], implemented in RecBole [28]. These two collaborative filtering approaches are widely used and differ in architecture, offering a meaningful testbed for evaluating `CaliTune`. While `BPR` learns separate embeddings for each user and item, `DMF` uses shared multi-layer projection networks for users and items.

For each dataset, we conduct a parameter grid search in the following ranges: learning rate in [0.01, 0.005, 0.001, 0.0005, 0.0001], embedding size in [32, 64, 128, 256] and for `DMF` hidden layers in {[32, 32], [64, 64], [128, 128], [256, 256], [64, 64, 64, 64], [256, 128, 64, 64]}. For both datasets and models, the best performing embedding size is 256, and hidden layers of size [256, 256] for `DMF`. The best performing learning rate is 0.0005 for AMBAR and 0.0001 for ML-1M. All models are trained on 80-10-10 data splits.

Calibration hyper-parameters. For `CaliTune`, we explore candidate set sizes $m_{cand} \in [25, 50, 100, 200, 400]$ (plus 1600 for AMBAR), calibration weight $\lambda \in [0.0, 0.1, 0.25, 0.5, 0.75, 1.0]$, and learning rate in [1e-3, 5e-4, 1e-4] (plus 1e-5 for `DMF`). Fine-tuning is performed on a single instance of `BPR` and `DMF` per dataset, using batch size 16, learning rate decay every 20 epochs ($\gamma = 0.9$), early-stopping with patience of 40, and a maximum of 500 epochs. For CP, we follow prior work [3, 9, 14], varying λ_{CP} from 0.1 to 0.9 with step size 0.1, and setting the re-ranking depth m to 100 items.

We save the intermediate states of the recommenders as they are gradually fine-tuned with `CaliTune`. After the procedure terminates, we evaluate the *JSD* and *nDCG* of every saved state on the test set, allowing us to track the trade-off dynamics over time. For CP, we similarly evaluate recommendations for each value of λ_{CP}.

User groups. To better understand the effectiveness of our approach, we evaluate performance separately for three user groups. Following previous work [1, 14], users are ranked by the proportion of *HighPop* items in their profiles: the top 20% are labeled *Mainstream-focused* (as most interested in popular items), the bottom 20% as *Niche-focused* users (least interested in popular items), and the remaining 60% as *Diverse* users.

4 Results

To evaluate how well `CaliTune` matches the accuracy-calibration trade-off of the post-processing CP (**RQ1**), we analyze Figure 1[3].

[3]Due to space constraints, we present a representative selection of experiments that highlight the effectiveness of `CaliTune` across various settings. Full experimental data and additional plots are available in the companion repository.

Each subplot corresponds to a specific backbone-dataset pair, with *JSD*@10 (lower is better) on the x-axis and *nDCG*@10 (higher is better) on the y-axis (thus the desired performance state in each plot is in the top left corner, corresponding to $JSD = 0$ and $nDCG = 1$). Backbone's original performance is marked as "**x**", while CP's results across different λ_{CP} values are shown as "+", darker symbols reflect stronger calibration emphasis. Colored lines represent CaliTune's trade-off path across λ values: from red at $\lambda = 0.0$ for pure accuracy objective, to blue at $\lambda = 1.0$ for pure calibration objective, with more saturated dots indicating later fine-tuning epochs. For visual clarity, only epochs showing meaningful changes from the previous epoch are displayed. Learning rate and candidate number m_{cand} are noted in grey at the top of each plot. Additionally, for each setting, we report the CaliTune model that achieves the best calibration while keeping *nDCG* at or above[4] the closest CP point. We express CaliTune's relative calibration improvement as a percentage of the total improvement achieved by CP at $\lambda_{CP} = 0.9$.

To assess the sensitivity of CaliTune across different hyper-parameter settings and user groups (**RQ2**), we refer to Figure 2 and Figure 3. Figure 2 shows the trade-offs achieved by CaliTune when optimizing solely for accuracy ($\lambda = 0.0$) or calibration ($\lambda = 1.0$) on the AMBAR dataset for BPR. Labels (such as $m400$) indicate the number of top-m_{cand} candidates used during fine-tuning. The left plots show results when all embeddings are fine-tuned, while the right plots restrict updates to user embeddings only. Figure 3 breaks down CaliTune's performance over BPR by user type, highlighting differences across $Niche$, $Diverse$, $Mainstream$ users.

4.1 RQ1: Effectiveness

From Figure 1, we see several cases where CaliTune matches or outperforms the post-processing baselines in terms of accuracy-calibration trade-off. On ML-1M with BPR, CaliTune performs on par with CP at $\lambda_{CP} = 0.1$, achieving 31% of CP's total calibration without a significant drop in *nDCG*. On AMBAR, CaliTune calibrates 74% of CP's total, significantly improving *nDCG* from CP at $\lambda_{CP} = 0.4$. For DMF on ML-1M CaliTune reaches 61% of CP's maximum calibration, losing only 8.3% in *nDCG* compared to CP at $\lambda_{CP} = 0.3$. On AMBAR, CaliTune matches CP's trade-off at $\lambda_{CP} = 0.9$ with no significant *nDCG* difference. When prioritizing calibration over accuracy, CaliTune consistently achieves over 65% of CP's total calibration across all backbone-dataset combinations, albeit sometimes at a less favorable trade-off.

Key findings. Our experiments suggest that CaliTune can serve as a competitive, inference-efficient alternative to CP across different backbone-dataset combinations. Notably, the optimal hyper-parameters settings vary across these combinations, and further experiments are needed to assess whether CaliTune can consistently match CP under stronger calibration objectives.

4.2 RQ2: Sensitivity

As shown in Figure 1, combining accuracy and calibration objectives generally produces a smoother trade-off across all settings, which gradually changes with the λ adjustment. For BPR, fine-tuning yields a trade-off similar to CP, with a gradual decay of accuracy for all λ. In contrast, on DMF it behaves differently, with *nDCG* dropping at early epochs and recovering in later epochs, particularly on ML-1M. We attribute this to DMF's shared parameter structure for user and item embeddings while BPR learns independent embeddings for each entity, allowing for more flexible calibration to individual user preferences.

Figure 2 suggests that higher learning rates can enhance calibration, though finer adjustments may yield more balanced trade-off (e.g., *m200* with $\lambda = 1.0$ in the first two plots). Fine-tuning only user embeddings (two rightmost plots) results in more predictable trade-offs, allowing even the accuracy objective to improve calibration. Conversely, fine-tuning all embeddings can lead to better calibration trade-offs under certain settings (e.g., $lr = 1e-4$, *m200*). We also observe that while accuracy benefits from larger m_{cand}, calibration peaks at $m_{cand} <= 200$ and declines with more candidates. This indicates that CaliTune's objectives, while aligned, may reach their optima under different hyper-parameter settings.

Figure 3 shows that fine-tuning BPR results in more favorable trade-offs for user-groups initially exhibiting higher miscalibration: *Diverse* and *Mainstream-focused* on ML-1M, and *Niche* and *Diverse* on AMBAR. We compared CaliTune and CP to the backbone model in relation to the popularity of recommendations. On ML-1M, CaliTune reduced the number of popular items for nearly twice as many users as CP among Mainstream and Niche users. Moreover, CaliTune improved nDCG for 24–32% and JSD for up to 87% of users on ML-1M, highlighting its ability to support users with non-mainstream preferences without sacrificing accuracy. Similar trends were seen in AMBAR[5]. This suggests that CaliTune can be particularly effective for models like BPR with separate user and item embeddings, as it enables smoother trade-offs, and supports user- or group-specific fine-tuned representations, enhancing overall calibration.

Key findings. Through various hyper-parameter settings, CaliTune consistently achieves a wide range of accuracy-calibration trade-offs, especially in models with user-specific embeddings. It is particularly effective for improving calibration among initially more miscalibrated user groups.

5 Discussion

This work shows that fine-tuning can present a viable alternative to post-processing for improving item popularity calibration in recommender systems. This section discusses the experimental results and provides some practical considerations.

The most prominent advantage of CaliTune in its current state is zero added latency during inference, compared to the unchanged backbone (the fine-tuned model fully retains its original architecture and is deployed in the same way). In contrast, post-processing assumes changes to the inference pipeline and unavoidable added latency while producing recommendations. In case of CaliTune, the absence of added latency comes at the cost of additional efforts before deployment. Beyond fine-tuning itself, its hyper-parameters (such as candidate count and the balancing coefficient λ) need to be individually chosen for every backbone and dataset combination. Furthermore, our preliminary work shows that CaliTune provides comparable or better accuracy-calibration trade-offs only up to a

[4]Significantly better performance of CaliTune or no significant difference according to a Wilcoxon test with $p < 0.01$.

[5]We share additional analysis in the companion repository.

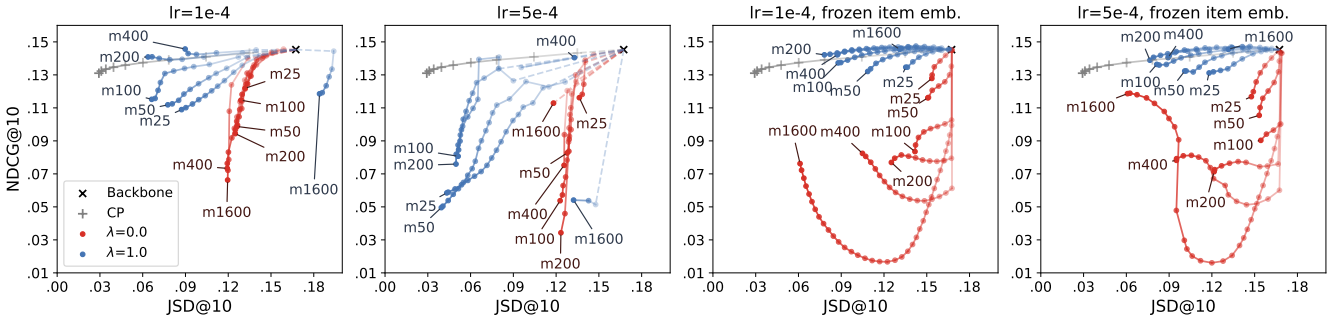

Figure 2: Accuracy–calibration trade-off for `CaliTune` on AMBAR with `BPR` as backbone, including accuracy ($\lambda = 0.0$) and calibration ($\lambda = 1.0$) objectives. We vary the candidate numbers (m_{cand}), learning rates and fine-tuned components (i.e. only user embeddings vs. full model). Fine-tuning of only user embeddings yields more stable trade-offs, while full fine-tuning achieves stronger calibration at potential accuracy cost.

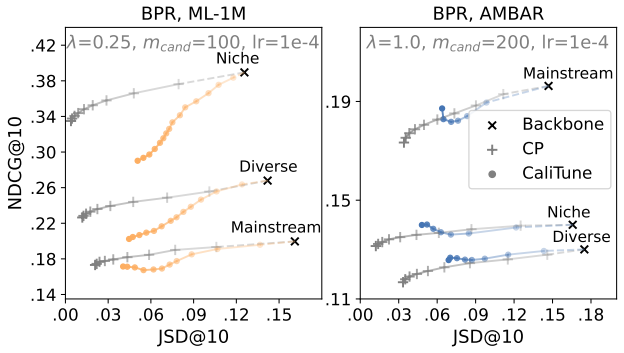

Figure 3: Accuracy-calibration trade-off of `CaliTune` for different user-groups on two datasets, with `BPR` as backbone. `CaliTune` is more effective for user-groups initially catered with higher miscalibration by the backbone model.

certain point after which CP performs more favorably. This way `CaliTune` is an attractive option in cases when inference time is more valuable than training time, given the resulting trade-off is sufficient for the task. Moreover, as `CaliTune` incorporates calibration objectives directly into the model's learning process, it adapts its internal representations to yield more robust results when the calibration target is not separable at ranking time.

This study shows feasibility of employing fine-tuning to improve calibration of already established recommenders. Yet, more research is needed to evaluate the efficiency of `CaliTune` for different types of attributes, beyond popularity. One would expect calibration of more fine-grained attributes to be more challenging to achieve through fine-tuning, as it requires capturing more item attribute categories by means of the fine-tuned model's parameters. Furthermore, some attributes may be easier to calibrate for, depending on the amount of information about them already present in the unchanged backbone.

Finally, we note directions for further improvement of `CaliTune`. The accuracy-oriented loss term hasn't performed equally well for all datasets, which motivates a search for an alternative formulation. The accuracy and calibration oriented loss terms show peak effectiveness at different candidate counts. Setting this parameter individually for each may improve the resulting trade-off.

6 Conclusion and future work

We introduce `CaliTune`, a fine-tuning approach for recommendation calibration. Unlike SotA methods, `CaliTune` neither incurs any additional computational load at the inference time, nor requires altering the backbone. We benchmark `CaliTune` against *Calibrated Popularity*, a leading post-processing technique on the same backbones. Our results show that `CaliTune` achieves comparable performance when the calibration effort prioritizes accuracy and, in some cases, is able to provide a superior accuracy-calibration trade-off, making it a viable inference-efficient alternative.

Beyond its immediate results, this work aims to spark broader discussion around fine-tuning as a general strategy for calibration. There is promising potential to improve the accuracy-calibration trade-off of `CaliTune`, including personalization of calibration intensity, more flexible multi-objective optimization and smarter sampling mechanisms. For example, `CaliTune`'s reliance on top-m_{cand} candidates for fine-tuning can limit its ability to promote relevant but initially low-ranked items. This could be mitigated by hybrid sampling strategies that also consider historically relevant items. Moreover, in future work we plan to investigate the efficiency of `CaliTune` in conjunction with more modern recommendation architectures[6], as well as applied to other finer-grained attributes (e.g., genre, recency) or personalized semantic clusters, and domains. While additional research is needed to fully realize these possibilities, we see `CaliTune` as an important step towards more integrated approaches to calibrated recommendation.

Acknowledgments

This research was funded in whole or in part by the Austrian Science Fund (FWF): https://doi.org/10.55776/COE12, https://doi.org/10.55776/P36413. This work received financial support from the State of Upper Austria and the Federal Ministry of Education, Science, and Research, through grant LIT-2024-13-SEE-111.

[6] Our preliminary results obtained on MultiVAE can be found in the repository.

References

[1] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. 2019. The Unfairness of Popularity Bias in Recommendation. In *Proceedings of the Workshop on Recommendation in Multi-stakeholder Environments co-located with the 13th ACM Conference on Recommender Systems, Copenhagen, Denmark, September 20, 2019 (CEUR Workshop Proceedings, Vol. 2440)*.

[2] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, and Bamshad Mobasher. 2020. The Connection Between Popularity Bias, Calibration, and Fairness in Recommendation. In *Proceedings of the 14th ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020*. 726–731.

[3] Himan Abdollahpouri, Masoud Mansoury, Robin Burke, Bamshad Mobasher, and Edward C. Malthouse. 2021. User-centered Evaluation of Popularity Bias in Recommender Systems. In *Proceedings of the 29th ACM Conference on User Modeling, Adaptation and Personalization, UMAP 2021, Utrecht, The Netherlands, June, 21-25, 2021*. 119–129.

[4] Himan Abdollahpouri, Zahra Nazari, Alex Gain, Clay Gibson, Maria Dimakopoulou, Jesse Anderton, Benjamin A. Carterette, Mounia Lalmas, and Tony Jebara. 2023. Calibrated Recommendations as a Minimum-Cost Flow Problem. In *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, WSDM 2023, Singapore, 27 February 2023 - 3 March 2023*. ACM, 571–579.

[5] Jiayi Chen, Wen Wu, Liye Shi, Yu Ji, Wenxin Hu, Xi Chen, Wei Zheng, and Liang He. 2022. DACSR: Decoupled-Aggregated End-to-End Calibrated Sequential Recommendation. *Applied Sciences* 12, 22 (2022).

[6] Elizabeth Gómez, David Contreras, Ludovico Boratto, and Maria Salamó. 2024. AMBAR: A dataset for Assessing Multiple Beyond-Accuracy Recommenders. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys 2024, Bari, Italy, October 14-18, 2024*. ACM, 137–147.

[7] F. Maxwell Harper and Joseph A. Konstan. 2016. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.* 5, 4 (2016), 19:1–19:19.

[8] Hyunsik Jeon, Se-eun Yoon, and Julian J. McAuley. 2024. Calibration-Disentangled Learning and Relevance-Prioritized Reranking for Calibrated Sequential Recommendation. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management, CIKM 2024, Boise, ID, USA, October 21-25, 2024*. ACM, 973–982.

[9] Mesut Kaya and Derek G. Bridge. 2019. A comparison of calibrated and intent-aware recommendations. In *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*. ACM, 151–159.

[10] Jon M. Kleinberg, Emily Ryu, and Éva Tardos. 2024. Calibrated Recommendations for Users with Decaying Attention. In *Algorithmic Game Theory - 17th International Symposium, SAGT 2024, Amsterdam, The Netherlands, September 3-6, 2024, Proceedings (Lecture Notes in Computer Science, Vol. 15156)*. Springer, 443–460.

[11] Anastasiia Klimashevskaia, Mehdi Elahi, Dietmar Jannach, Lars Skjærven, Astrid Tessem, and Christoph Trattner. 2023. Evaluating The Effects of Calibrated Popularity Bias Mitigation: A Field Study. In *Proceedings of the 17th ACM Conference on Recommender Systems, RecSys 2023, Singapore, Singapore, September 18-22, 2023*. ACM, 1084–1089.

[12] Anastasiia Klimashevskaia, Mehdi Elahi, Dietmar Jannach, Christoph Trattner, and Lars Skjærven. 2022. Mitigating Popularity Bias in Recommendation: Potential and Limits of Calibration Approaches. In *Advances in Bias and Fairness in Information Retrieval - Third International Workshop, BIAS 2022, Stavanger, Norway, April 10, 2022, Revised Selected Papers (Communications in Computer and Information Science, Vol. 1610)*. 82–90.

[13] Anastasiia Klimashevskaia, Dietmar Jannach, Mehdi Elahi, and Christoph Trattner. 2024. A survey on popularity bias in recommender systems. *User Model. User Adapt. Interact.* 34, 5 (2024), 1777–1834.

[14] Oleg Lesota, Stefan Brandl, Matthias Wenzel, Alessandro B. Melchiorre, Elisabeth Lex, Navid Rekabsaz, and Markus Schedl. 2022. Exploring Cross-group Discrepancies in Calibrated Popularity for Accuracy/Fairness Trade-off Optimization. In *Proceedings of the 2nd Workshop on Multi-Objective Recommender Systems co-located with 16th ACM Conference on Recommender Systems (RecSys 2022), Seattle, WA, USA, 18th-23rd September 2022 (CEUR Workshop Proceedings, Vol. 3268)*. CEUR-WS.org.

[15] Oleg Lesota, Jonas Geiger, Max Walder, Dominik Kowald, and Markus Schedl. 2024. Oh, Behave! Country Representation Dynamics Created by Feedback Loops in Music Recommender Systems. In *Proceedings of the 18th ACM Conference on Recommender Systems, RecSys 2024, Bari, Italy, October 14-18, 2024*. ACM, 1022–1027.

[16] Oleg Lesota, Alessandro B. Melchiorre, Navid Rekabsaz, Stefan Brandl, Dominik Kowald, Elisabeth Lex, and Markus Schedl. 2021. Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected?. In *Proceedings of the 15th ACM Conference on Recommender Systems (Late-Breaking Results)*. Amsterdam, the Netherlands.

[17] J. Lin. 1991. Divergence measures based on the Shannon entropy. *IEEE Transactions on Information Theory* 37, 1 (1991), 145–151.

[18] Kun Lin, Masoud Mansoury, Farzad Eskandanian, Milad Sabouri, and Bamshad Mobasher. 2024. Beyond Static Calibration: The Impact of User Preference Dynamics on Calibrated Recommendation. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization, UMAP Adjunct 2024, Cagliari, Italy, July 1-4, 2024*. ACM. doi:10.1145/3631700.3664869

[19] Kun Lin, Nasim Sonboli, Bamshad Mobasher, and Robin Burke. 2020. Calibration in Collaborative Filtering Recommender Systems: a User-Centered Analysis. In *HT '20: 31st ACM Conference on Hypertext and Social Media, Virtual Event, USA, July 13-15, 2020*. ACM, 197–206.

[20] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *UAI 2009, Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence, Montreal, QC, Canada, June 18-21, 2009*. AUAI Press, 452–461.

[21] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian Personalized Ranking from Implicit Feedback. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence* (Montreal, Quebec, Canada) *(UAI '09)*. AUAI Press, Arlington, Virginia, USA, 452–461.

[22] Andre Sacilotti, Rodrigo Ferrari de Souza, and Marcelo G. Manzato. 2023. Counteracting Popularity-Bias and Improving Diversity Through Calibrated Recommendations. In *Proceedings of the 25th International Conference on Enterprise Information Systems, ICEIS 2023, Volume 1, Prague, Czech Republic, April 24-26, 2023*. SCITEPRESS, 709–720.

[23] Sinan Seymen, Himan Abdollahpouri, and Edward C. Malthouse. 2021. A Constrained Optimization Approach for Calibrated Recommendations. In *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*. ACM, 607–612.

[24] Pannagadatta Shivaswamy. 2023. ExCalibR: Expected Calibration of Recommendations. *CoRR* abs/2304.12311 (2023). arXiv:2304.12311

[25] Harald Steck. 2018. Calibrated recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*. ACM, 154–162.

[26] Hong-Jian Xue, Xinyu Dai, Jianbing Zhang, Shujian Huang, and Jiajun Chen. 2017. Deep Matrix Factorization Models for Recommender Systems. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*. ijcai.org, 3203–3209.

[27] George Zerveas, Navid Rekabsaz, Daniel Cohen, and Carsten Eickhoff. 2022. CODER: An efficient framework for improving retrieval through COntextual Document Embedding Reranking. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*. Association for Computational Linguistics, 10626–10644.

[28] Wayne Xin Zhao, Shanlei Mu, Yupeng Hou, Zihan Lin, Yushuo Chen, Xingyu Pan, Kaiyuan Li, Yujie Lu, Hui Wang, Changxin Tian, Yingqian Min, Zhichao Feng, Xinyan Fan, Xu Chen, Pengfei Wang, Wendi Ji, Yaliang Li, Xiaoling Wang, and Ji-Rong Wen. 2021. RecBole: Towards a Unified, Comprehensive and Efficient Framework for Recommendation Algorithms. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management* (Virtual Event, Queensland, Australia) *(CIKM '21)*. Association for Computing Machinery, New York, NY, USA, 4653–4664.