The Impact of Differential Privacy on Recommendation Accuracy and Popularity Bias

Peter Müllner[1][0000−0001−6581−1945], Elisabeth Lex[2][0000−0001−5293−2967], Markus Schedl[3,4][0000−0003−1706−3406], and Dominik Kowald[1,2][0000−0003−3230−6234]

[1] Know-Center GmbH, Graz, Austria
{pmuellner, dkowald}@know-center.at
[2] Graz University of Technology, Graz, Austria
elisabeth.lex@tugraz.at
[3] Johannes Kepler University Linz, Linz, Austria
markus.schedl@jku.at
[4] Linz Institute of Technology, Linz, Austria

Abstract. Collaborative filtering-based recommender systems leverage vast amounts of behavioral user data, which poses severe privacy risks. Thus, often random noise is added to the data to ensure Differential Privacy (DP). However, to date it is not well understood in which ways this impacts personalized recommendations. In this work, we study how DP affects recommendation accuracy and popularity bias when applied to the training data of state-of-the-art recommendation models. Our findings are three-fold: First, we observe that nearly all users' recommendations change when DP is applied. Second, recommendation accuracy drops substantially while recommended item popularity experiences a sharp increase, suggesting that popularity bias worsens. Finally, we find that DP exacerbates popularity bias more severely for users who prefer unpopular items than for users who prefer popular items.

Keywords: Recommender Systems · Collaborative Filtering · Differential Privacy · Accuracy · Popularity Bias.

1 Introduction

Modern collaborative filtering-based recommender systems aim to generate personalized recommendations that cater to the specific preferences of each individual user. Such recommender systems need to provide recommendations of high accuracy and must ensure that the recommendations do not exhibit popularity bias, i.e., overestimate the relevance of popular items. For this, vast amounts of user data need to be processed, which exposes the users to many severe privacy risks [6,46,54,9], e.g., the disclosure of rating data [9] or the inference of sensitive user attributes [49,19]. Thus, besides recommendation accuracy and popularity bias, user privacy is another important aspect of recommender systems research. Hence, it is critical to leverage privacy-preserving techniques such as *Differential Privacy (DP)* [13] to devise privacy-aware recommender systems.

Many mechanisms utilized to establish DP include the injection of random noise into the users' interaction data, which typically decreases the overall recommendation accuracy [7,56]. For recommender systems, some widely used mechanisms are the Gaussian or Laplacian Input Perturbation [17,13], Plausible Deniability [36,15], or the 1-Bit mechanism [12,10]. In particular, the 1-Bit mechanism is a natural match to the binary feedback data prevalent in modern recommender systems. This mechanism randomly substitutes parts of the positive feedback data with negative or missing feedback data, and then, this modified data is used to train the recommendation model. Specifically, the amount of positive feedback data that is randomly substituted depends on the privacy budget ϵ, i.e., a hyperparameter that controls how much random noise is incorporated into the recommendation process and what level of DP is achieved.

However, how DP impacts personalized recommendations is not well understood. Specifically, it remains unclear whether DP impacts the recommendations of all users, or just some users, and research on the connection between the ϵ value and the drop in recommendation accuracy is scarce. Also, how DP and the ϵ value impact the item popularity distribution in the respective recommendation lists and thus, popularity bias, is an open research topic. To shed light on these issues, in this work, we address the following three research questions:

- *How many users are impacted by DP? (RQ1)*
- *How does the privacy budget ϵ influence the accuracy drop? (RQ2)*
- *In which ways does DP impact popularity bias? (RQ3)*
  a. *How does DP impact the popularity distribution of the recommendations?*
  b. *How does DP impact popularity bias for different user groups?*

Accordingly, we perform experiments with a neural matrix-factorization model (i.e., *ENMF* [11]), a graph convolution network model (i.e., *LightGCN* [23]), and a variational autoencoder model (i.e., *MultVAE* [30]), and use datasets from the movie (i.e., *MovieLens 1M* [21]), music streaming (i.e., *LastFM User Groups* [27]), and online retail domain (i.e., *Amazon Grocery & Gourmet* [40]). Plus, we test various ϵ values to cater for different levels of privacy.

Our results show that nearly all users are impacted by DP, i.e., their recommendations are different from those generated without DP. Plus, this difference increases when ϵ becomes smaller (*RQ1*). With respect to recommendation accuracy, we find that DP leads to a substantial drop, which is most severe for small ϵ values. This highlights the trade-off between recommendation accuracy and privacy (*RQ2*). Similarly, we present strong evidence that DP increases popularity bias, in particular, when ϵ becomes smaller. This underlines an important trade-off between popularity bias and privacy. Moreover, we identify a "the poor get poorer" effect: DP increases popularity bias, especially for users that are already prone to strong popularity bias without DP, i.e., users that prefer unpopular items (cf. the unfairness of popularity bias [1,27]).

Overall, this work extends existing research on the trade-off between recommendation accuracy and privacy, and contributes novel insights on the connection between DP and popularity bias.

2 Related Work

Several previous works [46,9,6,54] identified many critical privacy risks for users in collaborative filtering-based recommender systems. For example, through the recommendations, the recommender system could leak user data to malicious parties [9,50,22], or an adversary could infer sensitive attributes of the user, e.g., gender [49,55,19]. To address these privacy risks, privacy-enhancing techniques, such as *Federated Learning* [31,33], *Homomorphic Encryption* [20,24], or *Differential Privacy (DP)* [32,36,13] need to be incorporated into the recommender system. However, Homomorphic Encryption has high computational complexity, and Federated Learning can still leak sensitive user information [39,42].

Therefore, in the past years, DP has emerged as a prominent choice in the recommender systems research community. However, one important shortcoming of DP is its negative impact on recommendation accuracy: DP typically leads to a substantial accuracy drop, since it incorporates random noise into the recommendation process [7,18]. Several works address this trade-off between recommendation accuracy and privacy by applying DP in different ways [18,38], e.g., by applying DP only to parts of the dataset [51,36], or by carefully tuning the degree of noise [56]. In detail, Zhu et al. [56] monitor how strong the item-to-item similarities would change if a piece of user data was not present. This way, they can better estimate what minimal level of random noise is necessary to ensure DP, and increase recommendation accuracy over comparable approaches. In a recommender system, there are typically a few users that are willing to openly share their data and many users that are less inclined to share their data. Xin and Jaakkola [51] exploit this, and protect only a subset of users with DP, which facilitates recommendation accuracy. Müllner et al. [36] attach to this, and modify the recommendation process of user-based KNN to minimize the number of users to which DP needs to be applied.

Besides privacy, another critical problem of recommender systems is popularity bias, i.e., the recommender system overestimates the relevance of popular items and therefore, popular items are overrepresented in the recommendations [1]. This can be regarded as disadvantaged, or “unfair” treatment of users that prefer unpopular items, since the recommendations do not match these users as well as users that prefer popular items. In theory, DP and fairness are closely connected to each other [14,53], since for both, a user’s data needs to be hidden from the recommender system, either to preserve privacy, or to prohibit discrimination based on, e.g., age or gender. In practice, correlations in the dataset can still reveal age or gender and therefore, lead to unfairness [16,5]. In this vein, several works [16,3,52] investigate the trade-off between fairness and privacy. For example, Sun et al. [47] use user data that is protected with DP to rectify the recommendations to increase fairness. Similarly, also Yang et al. [52] use post-processing to optimize for fairness with respect to recommendation accuracy. They observe that regarding recommendation accuracy, DP can lead to more unfairness; however, they do not address DP’s impact on popularity bias.

Despite few existing works, how DP impacts personalized recommendations remains an understudied problem and many research gaps exist. For example, whether the recommendations of all users are impacted, or how beyond-accuracy objectives, such as reducing popularity bias or increasing diversity, are impacted. Thus, our work attaches to existing work with respect to studying DP's impact on recommendation accuracy, and in addition, we provide novel insights to DP's impact on the trade-off between privacy and popularity bias.

3 Method

In this section, we explain how DP is applied to the user data and then, we present multiple evaluation metrics to quantify DP's impact on recommendation accuracy and popularity bias. Also, we describe the datasets used in this study, and provide all preprocessing steps. Finally, we detail the experimental setup including the hyperparameters, recommendation models, and our precise evaluation protocol. We also provide our source-code to foster reproducibiltity.

3.1 Differential Privacy for Implicit Feedback

To ensure DP, we use the DP-mechanism from Ding et al. [12], which is a natural match to the binary implicit feedback data prevalent in today's recommender systems [10]. With this mechanism, for positive feedbacks $\mathcal{D}^+$ and negative or missing feedbacks $\mathcal{D}^-$ between users and items, the probability that a feedback $f_{u,i}$ between user u and item i is present in the DP dataset $\mathcal{D}^+_{\mathcal{DP}}$ is:

$$Pr[f_{u,i} \in \mathcal{D}^+_{DP}] = \begin{cases} \frac{e^\epsilon}{e^\epsilon+1} & \text{if } f_{u,i} \in \mathcal{D}^+ \\ \frac{1}{e^\epsilon+1} & \text{if } f_{u,i} \in \mathcal{D}^- \end{cases} \tag{1}$$

where ϵ is the privacy budget [13] (i.e., it quantifies how much privacy loss is tolerated; the higher, the less noise is added). In addition to positive feedback data, also negative or missing feedback data can be randomly added to $\mathcal{D}^+_{DP}$. However, the recommendation model is unable to identify these feedbacks and assumes that all feedbacks in $\mathcal{D}^+_{DP}$ are positive. By applying this mechanism to the training data of the recommendation model the recommendations shall not leak information about the data that has been used in the recommendation process. For computational efficiency, we follow Chen et al. [10] and randomly sample one negative feedback for each positive feedback.

3.2 Evaluation Metrics

To identify users that are impacted by DP, we compute the Jaccard distance between a user u's recommendation list $\mathcal{R}(u)$ generated without DP and u's recommendation list $\mathcal{R}_{DP}(u)$ generated with DP applied to the training data. For the recommendation lists, we use the common cut-off of $n = 10$ items. Formally, the set of users impacted by DP (i.e., $U_{impacted}$) is given by:

$$U_{impacted} = \left\{ u \in U : \frac{|\mathcal{R}(u) \cap \mathcal{R}_{DP}(u)|}{|\mathcal{R}(u) \cup \mathcal{R}_{DP}(u)|} > 0 \right\} \tag{2}$$

where U is the set of all users. This means that we consider a user u as impacted, if DP leads to at least one different item in u's recommendation list.

Overall, we quantify to what degree DP impacts recommendation accuracy and popularity bias of a user u's recommendations via measuring the relative change of an evaluation metric μ, when DP is applied (cf. [28,37]):

$$\text{Relative change } \mathit{\Delta}\mu(u) = \frac{\mu_{DP}(u) - \mu(u)}{\mu(u)} \tag{3}$$

$$\text{Average relative change } \mathit{\Delta}\mu = \frac{1}{|U_{impacted}|} \sum_{u \in U_{impacted}} \mathit{\Delta}\mu(u) \tag{4}$$

where $\mu_{DP}(u)$ is the value of the metric for user u when DP is applied and $\mu(u)$ is the value of the metric without applying DP. Furthermore, $\mathit{\Delta}\mu$ denotes the average change over all impacted users.

Recommendation Accuracy. To study the impact of DP on recommendation accuracy, we compute the ranking-agnostic *Recall* [41] metric. In this work, we use ranking-agnostic metrics since they fit to the way in which we identify impacted users, i.e., whether any item in the recommendation list changes due to DP, disregarding the ordering of the items within the recommendation list. We do not additionally include *Precision*, since $\mathit{\Delta Recall} = \mathit{\Delta Precision}$[5].

Popularity Bias. We evaluate DP's impact on popularity bias via measuring the *Average Recommendation Popularity (ARP)* [26], i.e., the average fraction of users that interacted with a recommended item:

$$ARP(u) = \frac{1}{|\mathcal{R}(u)|} \sum_{i \in \mathcal{R}(u)} \phi(i) \tag{5}$$

where $\mathcal{R}(u)$ is the recommendation list of user u, and item i's popularity $\phi(i) = |U_i|/|U|$ is the number of users that interacted with i, i.e., $|U_i|$, divided by the number of all users $|U|$. Several works suggest [1,27] that users that prefer unpopular items experience more popularity bias than users that prefer popular items. Thus, we use *Popularity Lift (PopLift)* [2] to quantify popularity bias for distinct user groups. Specifically, this metric indicates whether the ARP matches the average item popularity $\mathit{\Gamma}(\cdot)$ in the average user's profile of user group G:

$$PopLift(G) = \frac{\sum_{u \in G} ARP(u) - \sum_{u \in G} \mathit{\Gamma}(u)}{\sum_{u \in G} \mathit{\Gamma}(u)} \tag{6}$$

We inspect two user groups: users that prefer items of low popularity, i.e., U_{low}, and users that prefer items with high popularity, i.e., U_{high}. We follow Abdollahpouri et al. [1] and correspondingly define U_{low} as the set of the 20% of users

[5] The number of recommended relevant items is divided by the number of all relevant items (i.e., *Recall*), or by the length of the recommendation list (i.e., *Precision*). When DP is applied, $\mathit{\Delta Recall}$ and $\mathit{\Delta Precision}$ only depend on how the number of recommended relevant items changes and therefore, the relative change is the same.

Table 1: Descriptive statistics of the three datasets. *Users* is the number of users, *Items* is the number of items, *Interactions* is the amount of interactions in the dataset, i.e., positive feedback, *Profile Size* is the average number of interactions per user, and *Density* is the inverse sparsity of the dataset in percent.

Dataset	Users	Items	Interactions	Profile Size	Density
MovieLens 1M	6,038	3,533	575,281	95.28	2.70%
LastFM User Groups	2,999	78,799	348,437	116.18	0.15%
Amazon Grocery & Gourmet	3,222	6,839	72,176	22.40	0.33%

with the lowest fraction of popular items in their profile, and U_{high} as the set of the 20% of users whose profiles contain the highest fraction of popular items. The set of popular items is given by the 20% of items with the highest item popularity scores $\phi(i)$. In addition, we test whether there exists a *Disparate Impact* [34] of DP on U_{low} and U_{high}. Therefore, we measure the *Gap* [35], i.e., the absolute difference between the $PopLift$ values of the two user groups:

$$Gap = |PopLift(U_{low}) - PopLift(U_{high})| \tag{7}$$

3.3 Datasets

For our experiments, we use three datasets, i.e., *MovieLens 1M* [21], *LastFM User Groups* [27], and *Amazon Grocery & Gourmet* [40] (see Table 1). *MovieLens 1M* and *Amazon Grocery & Gourmet* comprise rating scores in the range of 1 to 5, whereas *LastFM User Groups* comprises listening events between users and music artists [29,45]. For consistency and comparability, we follow [44] and sum the listening events per artist, followed by scaling the resulting scores to the range of 1 to 5. For *MovieLens 1M* and *LastFM User Groups*, we perform 20-core user pruning, followed by discarding scores below the respective mean value, i.e., 3.58 for *MovieLens 1M* and 1.13 for *LastFM User Groups*. We follow common practice [48], and regard all scores below this threshold, as well as missing scores, as negative feedback. For *Amazon Grocery & Gourmet*, we additionally 5-core item pruning before filtering the scores according to a threshold of 4.24.

3.4 Evaluation Protocol

We split each user's data into 60% training data used for model training, 20% validation data used for hyperparameter tuning, and 20% test data used for evaluation. After hyperparameter tuning (see Section 3.5), to research the impact of DP on personalized recommendations, we add DP to the training data (see Equation 1) and retrain the recommendation models to calculate the evaluation metrics (see Section 3.2). Specifically, we retrain each model with five different random seeds and average the evaluation metrics to cope for random

Table 2: Model hyperparameters used in our experiments (learning rate α, dropout probability ρ, embedding dimensionality d, negative weight ω, L_2 regularization factor λ, number of propagation layers n, number of hidden units h).

Model	*MovieLens 1M*	*LastFM User Groups*	*Amazon Grocery & Gourmet*
ENMF	$\alpha = 0.01, \rho = 0.1, d = 32, \omega = 0.25$	$\alpha = 0.001, \rho = 0.25, d = 128, \omega = 0.25$	$\alpha = 0.001, \rho = 0.25, d = 64, \omega = 0.25$
LightGCN	$\alpha = 0.0001, d = 128, n = 1, \lambda = 0.0001$	$\alpha = 0.001, d = 128, n = 4, \lambda = 0.01$	$\alpha = 0.001, d = 128, n = 2, \lambda = 0.001$
MultVAE	$\alpha = 0.01, \rho = 0.5, d = 64, h = 800$	$\alpha = 0.001, \rho = 0.5, d = 128, h = 600$	$\alpha = 0.0001, \rho = 0.5, d = 128, h = 600$

fluctuations in the training process. We repeat this procedure for multiple privacy budget values, i.e., $\epsilon \in \{5, 4, 3, 2, 1, 0.1, 0.01\}$. To foster the reproducibility of our research, we publish our source code in an anonymous repository[6].

3.5 Recommendation Models and Parameter Settings

To cover different kinds of recommender systems, we experiment with a neural matrix-factorization model, i.e., *ENMF* [11], a graph convolution network model, i.e., *LightGCN* [23], and a variational autoencoder model, i.e., *MultVAE* [30].

- *ENMF* [11] is an efficient neural matrix-factorization model that does not leverage negative sampling. Instead, a negative weighting scheme is used, which benefits training efficiency and recommendation accuracy.
- *LightGCN* [23] is a lightweight graph convolution network, which, in contrast to more complex approaches, only uses neighborhood aggregation and does not include feature transformations or nonlinear activations.
- *MultVAE* [30] is a variational autoencoder that generates recommendations based on a multinomial likelihood. This way, it aims to mimic the generative process of implicit feedback data as prevalent in recommender systems.

For model training, we use Adam [25] to optimize the models for 5,000 epochs with a batch size of 4,096, and we employ an early stopping threshold of 50. We perform grid search for every model-dataset pair and determine the hyperparameters of the model with the highest *Recall* on the validation data (see Table 2). Note that hyperparameter tuning is performed on the original training data without DP. *LightGCN* requires negative samples and therefore, we sample one negative sample for each positive feedback uniformly at random. After a careful inspection, we find that with the given hyperparameters, *LightGCN* cannot produce personalized recommendations for *Amazon Grocery & Gourmet*. To solve this, we manually adapt the learning rate to 0.001 and the batch size to 1,024. In all experiments, the top 10 ranked items are recommended to each user.

4 Results

In this section, we present our results with respect to the three research questions. First, we measure for how many users the recommendations differ when DP is

[6] `https://anonymous.4open.science/r/ImpactOfDP-DB86`

Table 3: Absolute values of the evaluation metrics when no DP is used. This serves as baseline for our results in the remainder of this paper, which measure the relative change of the evaluation metrics when DP is applied. For calculating the metrics, we use all impacted users.

	MovieLens 1M			*LastFM User Groups*			*Amazon Grocery & Gourmet*		
Model	*Recall* ↑	*ARP* ↓	*PopLift* ↓	*Recall* ↑	*ARP* ↓	*PopLift* ↓	*Recall* ↑	*ARP* ↓	*PopLift* ↓
ENMF	0.1697	0.2172	0.7084	0.0971	0.0836	1.8816	0.0932	0.0180	0.5143
LightGCN	0.1669	0.1958	0.5405	0.0925	0.0800	1.7585	0.0836	0.0259	1.1796
MultVAE	0.1694	0.1990	0.5657	0.0835	0.0576	0.9864	0.0643	0.0199	0.6734

applied, and we measure how strong these differences are (*RQ1*). Second, we detail these differences with respect to the relative change of recommendation accuracy (*RQ2*) and popularity bias (*RQ3a*). Plus, we investigate the impact of DP on popularity bias from the perspective of two user groups: users that prefer unpopular items and users that prefer popular items (*RQ3b*). As a baseline, Table 3 includes the absolute values of our evaluation metrics without DP.

4.1 Differences Between Recommendations

First, we approach *RQ1* and quantify how many users are impacted by DP (see Table 4). We find that for all datasets, recommendation models, and ϵ values, DP impacts nearly all users, i.e., different items are recommended than without DP. For these impacted users, the average difference, i.e., the Jaccard distance between the recommendations with and without DP, lies above 0.5. Thus, on average, more than every second item in the recommendation list would not have been recommended without DP. Overall, the impact of DP increases when ϵ becomes smaller, i.e., when more noise is added to the training data of the recommendation models. Specifically, for $\epsilon = 0.1$ and across all recommendation models and datasets, more than 99.99% of users are impacted by DP, and the average Jaccard distance lies between 0.8058 and 0.9743.

This gives strong evidence that DP fundamentally impacts the generated recommendations for nearly all users (RQ1).

4.2 Impact on Recommendation Accuracy

Next, we build on our results from *RQ1*, and study how DP's impact on the recommendation lists affects recommendation accuracy (*RQ2*). We find that DP leads to a substantial drop in recommendation accuracy, as measured by *Recall* (see Figure 1). In contrast to *MovieLens 1M* and *LastFM User Groups*, the recommendation accuracy for *Amazon Grocery & Gourmet* already drops in case $\epsilon = 5$, which is possibly due to DP being applied to the (on average) small user profiles in this dataset (see Table 1). In case of *ENMF* and *MultVAE*

Table 4: *No. Users* is the percentage of users that are impacted by DP and *Avg. J.* is the average Jaccard distance between the recommendations with and without DP. The worst results are given in **bold**. We find that nearly all users are impacted by DP and that the recommendations substantially differ from those generated without DP (*RQ1*).

		MovieLens 1M		*LastFM User Groups*		*Amazon Grocery & Gourmet*	
ϵ	Model	No. Users ↓	Avg. J. ↓	No. Users ↓	Avg. J. ↓.	No. Users ↓	Avg. J. ↓
5	*ENMF*	99.41%	0.5118	98.06%	0.4988	99.96%	0.7872
	LightGCN	97.40%	0.4207	99.14%	0.5112	99.94%	0.7382
	MultVAE	99.71%	0.5903	99.68%	0.6983	**100.00%**	0.9204
2	*ENMF*	99.85%	0.5974	99.64%	0.5757	**100.00%**	0.8620
	LightGCN	99.86%	0.6252	99.92%	0.6518	99.99%	0.8132
	MultVAE	99.93%	0.6828	**100.00%**	0.7950	**100.00%**	0.9447
1	*ENMF*	99.99%	0.7006	99.95%	0.6858	**100.00%**	0.9253
	LightGCN	**99.99%**	0.7352	99.99%	0.7464	**100.00%**	0.8775
	MultVAE	**100.00%**	0.7592	**100.00%**	0.8408	**100.00%**	0.9567
0.1	*ENMF*	**100.00%**	**0.8183**	**100.00%**	**0.8058**	**100.00%**	**0.9743**
	LightGCN	**99.99%**	**0.8300**	**100.00%**	**0.8490**	**100.00%**	**0.9360**
	MultVAE	**100.00%**	**0.8447**	**100.00%**	**0.9250**	**100.00%**	**0.9635**

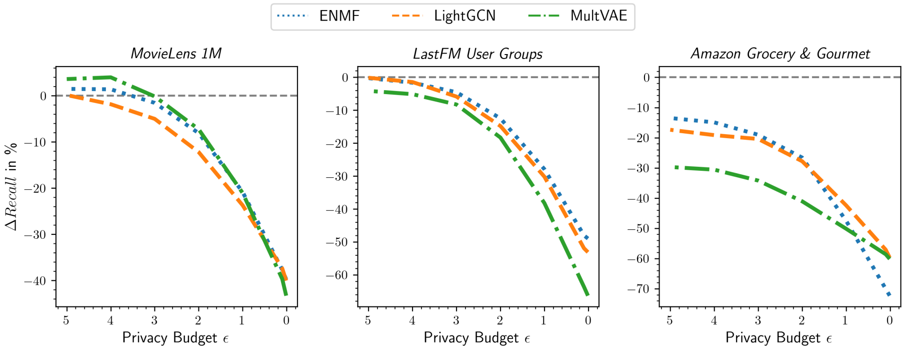


Fig. 1: DP's impact on recommendation accuracy as measured by *ΔRecall*. DP leads to a severe drop in recommendation accuracy. In particular, this drop becomes more serious for small ϵ values that provide a high level of privacy. This corresponds to the well-known accuracy-privacy trade-off (*RQ2*).

on *MovieLens 1M*, the recommendation accuracy increases slightly for large ϵ values, which can be possibly due to the fact that the noise introduced by DP acts as Tikhonov regularization for the model [8]. However, when more noise is added, i.e., $\epsilon < 3$, the recommendation accuracy for these models and dataset drops as well. Overall, the drop in recommendation accuracy gets worse when ϵ becomes smaller. Specifically, for $\epsilon = 0.1$ and across all recommendation models, the recommendation accuracy drops by at least 37.39% (*MovieLens 1M*), 48.00% (*LastFM User Groups*), or 57.10% (*Amazon Grocery & Gourmet*). Since lower ϵ values lead to higher levels of privacy, this corresponds to the well-known trade-off between recommendation accuracy and privacy [7,56].

In summary, DP leads to a substantial drop in recommendation accuracy, and this drop becomes more severe for smaller ϵ values (RQ2).

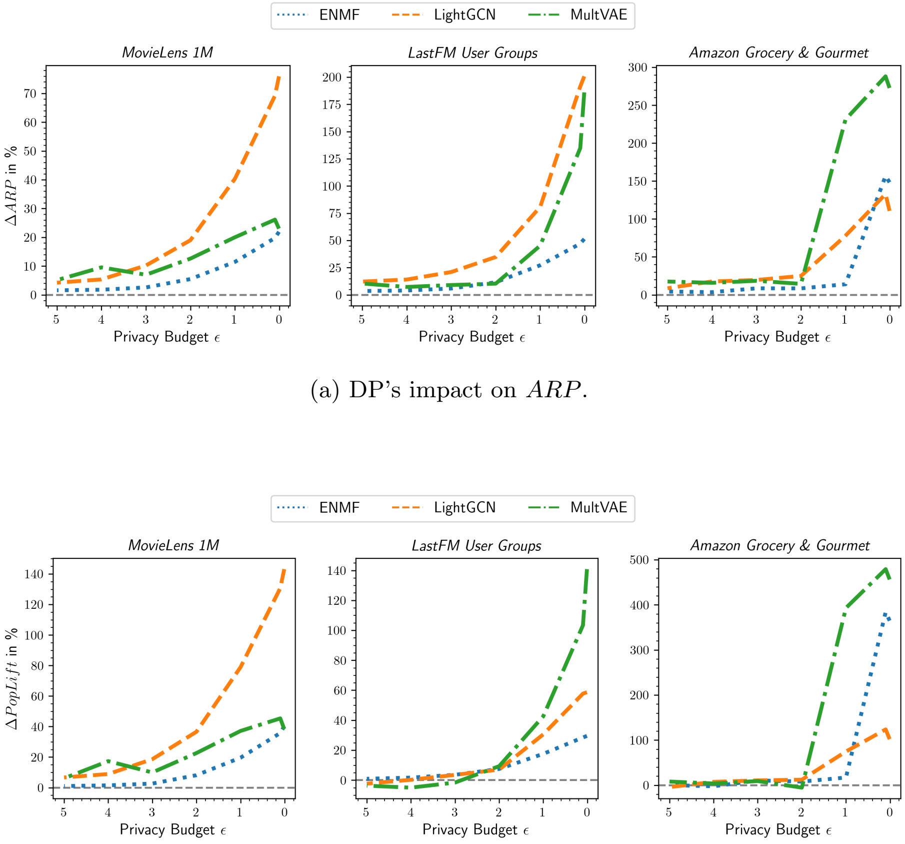
4.3 Impact on Popularity Bias

In this section, we study how DP impacts popularity bias (*RQ3*). First, in Figure 2, we monitor how DP impacts the average recommendation popularity (*ARP*) and the popularity lift (*PopLift*). Then, we investigate DP's impact on popularity bias from the perspective of two user groups: users that prefer unpopular items and users that prefer popular items.

Impact on Recommendation Popularity. We find that DP leads to a substantial increase with respect to *ARP* (see Figure 2a). Specifically, the increase in *ARP* gets worse, when ϵ becomes smaller. For example, for $\epsilon = 0.1$ and across all recommendation models, *ARP* increases by at least 19.75% (*MovieLens 1M*), 47.00% (*LastFM User Groups*), or 132.85% (*Amazon Grocery & Gourmet*). We investigate these differences in more detail, and find that the increase is especially high for datasets, for which the baseline value without DP is small (see Table 3). This means that without DP, also items of low popularity are recommended, which are typically hard to recommend (cf. the item cold-start problem [43]). With the noise introduced by DP, these items are even harder to recommend, which increases the *ARP* value. Thus, more popular items are recommended as ϵ becomes smaller, which indicates a trade-off between privacy and popularity bias. In adddition to *ARP*, we also use *PopLift* to quantify popularity bias, since it relates *ARP* to a user's preference for popular items (see Figure 2b). As in case of *ARP*, also *PopLift* increases when the ϵ value becomes smaller, i.e., the popularity of the recommended items increasingly mismatches the item popularity distribution in the users' profiles. Specifically, for $\epsilon = 0.1$ and across all recommendation models, *PopLift* increases by at least 36.16% (*MovieLens 1M*), 28.49% (*LastFM User Groups*), or 128.38% (*Amazon Grocery & Gourmet*). This means that as ϵ becomes smaller, there is an increasing mismatch between the recommendation popularity and the item popularity distribution of the users.

Therefore, DP makes the recommendations more biased towards popular items, which strongly overestimates the users' preferences for popular items. This underlines the important trade-off between privacy and popularity bias (RQ3a).

Disparate Impact on User Groups. Building upon our finding that DP makes popularity bias worse, we finally investigate whether the strength of this



(a) DP's impact on ARP.

(b) DP's impact on $PopLift$.

Fig. 2: DP's impact on popularity bias as measured by ΔARP and $\Delta PopLift$. We find that DP increases ARP, which becomes more severe the smaller the ϵ value is (see Figure 2a). Plus, the recommendation popularity mismatches the item popularity distribution in the user profiles (see Figure 2b). Overall, this gives strong evidence that DP makes popularity bias worse (*RQ3*).

effect differs between users that prefer popular items (i.e., U_{high}) and users that prefer unpopular items (i.e, U_{low}). For both user groups, *PopLift* increases for small ϵ values (see Table 5). Similarly, also the *Gap* between both user groups' $PopLift$ values grows when ϵ becomes smaller, which suggests that there exists a disparate impact of DP (cf. [52]). We investigate *Gap* and $PopLift$ in more detail and find that in general, *PopLift* increases more severely for U_{low} than for U_{high}. This can be regarded as a "poor get poorer" effect, since these disadvantaged users already experience strong popularity bias without DP. However, in case of *MultVAE* and *LastFM User Groups*, *PopLift* is higher for U_{high} than for

Table 5: The absolute *PopLift* values for users that prefer unpopular items (U_{low}) and users that prefer popular items (U_{high}), and the *Gap*, i.e., the absolute difference, between both. The worst results are given in **bold**. Popularity bias increases for both user groups with decreasing ϵ, but as *Gap* suggests, popularity bias increases especially for users that prefer unpopular, niche items (*RQ3b*).

		MovieLens 1M		*LastFM User Groups*		*Amazon* *Grocery & Gourmet*	
		PopLift ↓	*Gap* ↓	*PopLift* ↓	*Gap* ↓	*PopLift* ↓	*Gap* ↓
ϵ	Method	U_{low}/U_{high}		U_{low}/U_{high}		U_{low}/U_{high}	
No DP	*ENMF*	1.0923/0.4800	0.6124	4.1028/1.1578	2.9450	1.4079/0.0845	1.3235
	LightGCN	0.4225/0.5296	0.1072	2.7848/1.3273	1.4576	1.8253/0.7109	1.1144
	MultVAE	0.6247/0.4901	0.1347	0.7441/0.9092	0.1651	1.3910/0.2259	1.1650
5	*ENMF*	1.0940/0.4903	0.6037	4.0972/1.1629	2.9343	1.4043/0.0896	1.3147
	LightGCN	0.4625/0.5566	0.0940	2.7952/1.2790	1.5162	1.7539/0.6766	1.0773
	MultVAE	0.6538/0.5227	0.1311	0.7244/0.8928	0.1685	1.4372/0.2783	1.1589
2	*ENMF*	1.2088/0.5147	0.6941	4.5492/1.2334	3.3158	1.4977/0.1380	1.3597
	LightGCN	0.7447/0.6206	0.1241	3.1516/1.2894	1.8623	2.0951/0.7736	1.3215
	MultVAE	0.8409/0.5814	0.2595	0.1894/1.0524	0.8629	1.4175/0.2014	1.2161
1	*ENMF*	1.3658/0.5612	0.8046	5.2311/1.3309	3.9001	1.5723/0.1517	1.4206
	LightGCN	1.1633/0.7265	0.4368	3.8118/1.5267	2.2851	3.1031/1.2728	1.8303
	MultVAE	1.0044/0.6233	0.3811	0.3395/1.3139	**0.9744**	6.0433/2.0317	4.0117
0.1	*ENMF*	**1.5276/0.6445**	**0.8831**	**5.7217/1.4448**	**4.2769**	**4.9652/1.2375**	**3.7277**
	LightGCN	**1.7767/0.8415**	**0.9352**	**5.7233/1.6460**	**4.0773**	**4.5163/1.5600**	**2.9563**
	MultVAE	**1.1595/0.6370**	**0.5225**	**1.0760/1.7873**	0.7113	**7.5308/2.3216**	**5.2092**

U_{low}. It is known that for some datasets[7] *MultVAE* is able to recommend many unpopular items from the long-tail [4]. This results in lower *ARP* values than in case of the other datasets, i.e., 0.0184 for U_{low} and 0.0933 for U_{high} (without DP), which especially benefits U_{low}. Therefore, this helps to maintain a low *PopLift* value for U_{low}, and may explain why in this specific case, *PopLift* is lower for U_{low} than for U_{high}.

Overall, DP makes popularity bias worse for both user groups, but most severely for users that prefer unpopular items (RQ3b).

5 Conclusion and Future Work

In this work, we investigated in which ways Differential Privacy (DP) impacts personalized recommendations. In experiments with three datasets and three recommendation algorithms, we added DP to the training data of state-of-the-art recommendation models, and found that nearly all users' recommendations

[7] No clear pattern across datasets can be observed [4] and thus, this behavior of *MultVAE* needs to be researched in the future.

change when DP is applied. Also, for higher levels of privacy, recommendation accuracy drops substantially while popularity bias increases. In addition, we detail DP's impact on popularity bias and observe a "poor get poorer" effect: DP exacerbates popularity bias more severely for users who already experience strong popularity bias without DP, i.e., users who prefer unpopular items. Overall, our work further researches the trade-off between recommendation accuracy and privacy and in addition, provides novel insights on the important trade-off between popularity bias and privacy. In the future, we plan to research how users that are especially disadvantaged by DP, i.e., U_{low}, can reach a satisfactory trade-off between recommendation accuracy, popularity bias, and privacy. Specifically, we aim to test whether popularity bias mitigation strategies can help to prohibit the exacerbation of popularity bias for disadvantaged user groups.

Acknowledgments

This research is funded by the "DDAI" COMET Module within the COMET — Competence Centers for Excellent Technologies Programme, funded by the Austrian Federal Ministry for Transport, Innovation and Technology (bmvit), the Austrian Federal Ministry for Digital and Economic Affairs (bmdw), the Austrian Research Promotion Agency (FFG), the province of Styria (SFG) and partners from industry and academia. The COMET Programme is managed by FFG. Moreover, this research received support by the Austrian Science Fund (FWF): DFH-23 and P36413; and by the State of Upper Austria and the Federal Ministry of Education, Science, and Research, through grants LIT-2020-9-SEE-113. For open access purposes, the author has applied a CC BY public copyright license to any author accepted manuscript version arising from this submission.

References

1. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The unfairness of popularity bias in recommendation. Workshop on Recommendation in Multi-stakeholder Environments (RMSE), in conjunction with the 13th ACM Conference on Recommender Systems (RecSys) (2019)
2. Abdollahpouri, H., Mansoury, M., Burke, R., Mobasher, B.: The connection between popularity bias, calibration, and fairness in recommendation. In: Proceedings of the 14th ACM Conference on Recommender Systems (RecSys). pp. 726–731 (2020)
3. Agarwal, S.: Trade-offs between fairness, interpretability, and privacy in machine learning. Master's thesis, University of Waterloo (2020)
4. Anelli, V.W., Bellogín, A., Di Noia, T., Jannach, D., Pomo, C.: Top-n recommendation algorithms: A quest for the state-of-the-art. In: Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization (UMAP). pp. 121–131 (2022)
5. Bagdasaryan, E., Poursaeed, O., Shmatikov, V.: Differential privacy has disparate impact on model accuracy. In: Proceedings of the 33rd International Conference on Neural Information Processing Systems (NeurIPS). pp. 15479–15488 (2019)

6. Beigi, G., Liu, H.: A survey on privacy in social media: identification, mitigation, and applications. ACM Transactions on Data Science (TDS) **1**(1), 1–38 (2020)
7. Berkovsky, S., Kuflik, T., Ricci, F.: The impact of data obfuscation on the accuracy of collaborative filtering. Expert Systems with Applications **39**(5), 5033–5042 (2012)
8. Bishop, C.M.: Training with noise is equivalent to tikhonov regularization. Neural computation **7**(1), 108–116 (1995)
9. Calandrino, J.A., Kilzer, A., Narayanan, A., Felten, E.W., Shmatikov, V.: "you might also like:" privacy risks of collaborative filtering. In: 2011 IEEE Symposium on Security and Privacy (S&P). pp. 231–246 (2011)
10. Chen, C., Zhou, J., Wu, B., Fang, W., Wang, L., Qi, Y., Zheng, X.: Practical privacy preserving poi recommendation. ACM Transactions on Intelligent Systems and Technology (TIST) **11**(5), 1–20 (2020)
11. Chen, C., Zhang, M., Zhang, Y., Liu, Y., Ma, S.: Efficient neural matrix factorization without sampling for recommendation. ACM Transactions on Information Systems (TOIS) **38**(2), 1–28 (2020)
12. Ding, B., Kulkarni, J., Yekhanin, S.: Collecting telemetry data privately. In: Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS). pp. 3574–3583 (2017)
13. Dwork, C.: Differential privacy: A survey of results. In: International conference on theory and applications of models of computation (TAMC). pp. 1–19 (2008)
14. Dwork, C., Hardt, M., Pitassi, T., Reingold, O., Zemel, R.: Fairness through awareness. In: Proceedings of the 3rd innovations in theoretical computer science conference (ITCS). pp. 214–226 (2012)
15. Dwork, C., Roth, A., et al.: The algorithmic foundations of differential privacy. Now Publishers, Inc. (2014)
16. Ekstrand, M.D., Joshaghani, R., Mehrpouyan, H.: Privacy for all: Ensuring fair and equitable privacy protections. In: Proceedings of ACM Conference on Fairness, Accountability, and Transparency (FAccT). pp. 35–47 (2018)
17. Friedman, A., Berkovsky, S., Kaafar, M.A.: A differential privacy framework for matrix factorization recommender systems. User Modeling and User-Adapted Interaction (UMUAI) **26**(5), 425–458 (2016)
18. Friedman, A., Knijnenburg, B.P., Vanhecke, K., Martens, L., Berkovsky, S.: Privacy Aspects of Recommender Systems, pp. 649–688. Springer US, Boston, MA (2015). https://doi.org/10.1007/978-1-4899-7637-6_19"
19. Ganhör, C., Penz, D., Rekabsaz, N., Lesota, O., Schedl, M.: Unlearning protected user attributes in recommendations with adversarial training. In: Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). pp. 2142–2147. Springer (2022)
20. Gentry, C.: A fully homomorphic encryption scheme. Ph.D. thesis, Stanford university (2009)
21. Harper, F.M., Konstan, J.A.: The movielens datasets: History and context. ACM Transactions on Interactive Intelligent Systems (TiiS) **5**(4), 1–19 (2015)
22. Hashemi, H., Xiong, W., Ke, L., Maeng, K., Annavaram, M., Suh, G.E., Lee, H.H.S.: Data leakage via access patterns of sparse features in deep learning-based recommendation systems. Workshop on Trustworthy and Socially Responsible Machine Learning (TSRML), in conjunction with the 36th Conference on Neural Information Processing Systems (NeurIPS) (2022)
23. He, X., Deng, K., Wang, X., Li, Y., Zhang, Y., Wang, M.: Lightgcn: Simplifying and powering graph convolution network for recommendation. In: Proceedings of

the 43rd International ACM SIGIR conference on research and development in Information Retrieval (SIGIR). pp. 639–648. Springer (2020)

24. Kim, S., Kim, J., Koo, D., Kim, Y., Yoon, H., Shin, J.: Efficient privacy-preserving matrix factorization via fully homomorphic encryption. In: Proceedings of the 11th ACM on Asia conference on computer and communications security (ASIACCS). pp. 617–628 (2016)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proceedings of 3rd International Conference on Learning Representations (ICLR) (2015)
26. Klimashevskaia, A., Elahi, M., Jannach, D., Trattner, C., Skjærven, L.: Mitigating popularity bias in recommendation: Potential and limits of calibration approaches. In: Advances in Information Retrieval: Workshop on Algorithmic Bias in Search and Recommendation (BIAS) in conjunction with the 42nd European Conference on IR Research (ECIR). pp. 82–90. Springer (2022)
27. Kowald, D., Schedl, M., Lex, E.: The unfairness of popularity bias in music recommendation: A reproducibility study. In: Advances in Information Retrieval: 42nd European Conference on IR Research (ECIR). pp. 35–42. Springer (2020)
28. 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? In: Proceedings of the 15th ACM Conference on Recommender Systems (RecSys). pp. 601–606 (2021)
29. Lex, E., Kowald, D., Schedl, M.: Modeling popularity and temporal drift of music genre preferences. Transactions of the International Society for Music Information Retrieval **3**(1) (2020)
30. Liang, D., Krishnan, R.G., Hoffman, M.D., Jebara, T.: Variational autoencoders for collaborative filtering. In: Proceedings of the World Wide Web Conference (TheWebConf). pp. 689–698 (2018)
31. Lin, Y., Ren, P., Chen, Z., Ren, Z., Yu, D., Ma, J., Rijke, M.d., Cheng, X.: Meta matrix factorization for federated rating predictions. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR). pp. 981–990. Springer (2020)
32. Long, J., Chen, T., Nguyen, Q.V.H., Yin, H.: Decentralized collaborative learning framework for next poi recommendation. ACM Trans. Inf. Syst. **41**(3) (2023). https://doi.org/10.1145/3555374
33. McMahan, B., Moore, E., Ramage, D., Hampson, S., y Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS). pp. 1273–1282 (2017)
34. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A survey on bias and fairness in machine learning. ACM computing surveys (CSUR) **54**(6), 1–35 (2021)
35. Melchiorre, A.B., Rekabsaz, N., Parada-Cabaleiro, E., Brandl, S., Lesota, O., Schedl, M.: Investigating gender fairness of recommendation algorithms in the music domain. Information Processing & Management (IP&P) **58**(5), 102666 (2021)
36. Müllner, P., Lex, E., Schedl, M., Kowald, D.: Reuseknn: Neighborhood reuse for differentially-private knn-based recommendations. ACM Trans. Intell. Syst. Technol. (2023). https://doi.org/10.1145/3608481
37. Müllner, P., Kowald, D., Lex, E.: Robustness of meta matrix factorization against strict privacy constraints. In: Advances in Information Retrieval: 43rd European Conference on IR Research (ECIR). pp. 107–119. Springer (2021)

38. Müllner, P., Lex, E., Schedl, M., Kowald, D.: Differential privacy in collaborative filtering recommender systems: a review. Frontiers in Big Data **6** (2023). https://doi.org/10.3389/fdata.2023.1249997
39. Nasr, M., Shokri, R., Houmansadr, A.: Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In: Proceedings of the IEEE Symposium on Security and Privacy (S&P). pp. 739–753 (2019)
40. Ni, J., Li, J., McAuley, J.: Justifying recommendations using distantly-labeled reviews and fine-grained aspects. In: Proceedings of the conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP). pp. 188–197 (2019)
41. Parra, D., Sahebi, S.: Recommender systems: Sources of knowledge and evaluation metrics. In: Advanced Techniques in Web Intelligence-2: Web User Browsing Behaviour and Preference Analysis, pp. 149–175. Springer (2013)
42. Ren, H., Deng, J., Xie, X.: Grnn: Generative regression neural network—a data leakage attack for federated learning. ACM Transactions on Intelligent Systems and Technology (TIST) **13**(4), 1–24 (2022)
43. Saveski, M., Mantrach, A.: Item cold-start recommendations: learning local collective embeddings. In: Proceedings of the 8th ACM Conference on Recommender systems (RecSys). pp. 89–96 (2014)
44. Schedl, M., Bauer, C.: Distance-and rank-based music mainstreaminess measurement. In: Adjunct publication of the 25th conference on user modeling, adaptation and personalization (UMAP): Workshop on Surprise, Opposition, and Obstruction in Adaptive and Personalized Systems (SOAP). pp. 364–367 (2017)
45. Schedl, M., Bauer, C., Reisinger, W., Kowald, D., Lex, E.: Listener modeling and context-aware music recommendation based on country archetypes. Frontiers in Artificial Intelligence **3**, 508725 (2021)
46. Shyong, K., Frankowski, D., Riedl, J., et al.: Do you trust your recommendations? an exploration of security and privacy issues in recommender systems. In: International Conference on Emerging Trends in Information and Communication Security (ETRICS). pp. 14–29. Springer (2006)
47. Sun, J.A., Pentyala, S., Cock, M.D., Farnadi, G.: Privacy-preserving fair item ranking. In: European Conference on Information Retrieval (ECIR). pp. 188–203. Springer (2023)
48. Sun, Z., Yu, D., Fang, H., Yang, J., Qu, X., Zhang, J., Geng, C.: Are we evaluating rigorously? benchmarking recommendation for reproducible evaluation and fair comparison. In: Proceedings of the 14th ACM Conference on Recommender Systems (RecSys). pp. 23–32 (2020)
49. Weinsberg, U., Bhagat, S., Ioannidis, S., Taft, N.: Blurme: Inferring and obfuscating user gender based on ratings. In: Proceedings of the 6th ACM conference on Recommender systems (RecSys). pp. 195–202 (2012)
50. Xin, X., Yang, J., Wang, H., Ma, J., Ren, P., Luo, H., Shi, X., Chen, Z., Ren, Z.: On the user behavior leakage from recommender system exposure. ACM Transactions on Information Systems (TOIS) **41**(3), 1–25 (2023)
51. Xin, Y., Jaakkola, T.: Controlling privacy in recommender systems. In: Proceedings of the 27th International Conference on Neural Information Processing Systems (NeurIPS). pp. 2618–2626. MIT Press, Cambridge, MA, USA (2014)
52. Yang, Z., Ge, Y., Su, C., Wang, D., Zhao, X., Ying, Y.: Fairness-aware differentially private collaborative filtering. In: Companion Proceedings of the ACM Web Conference (TheWebConf). pp. 927–931 (2023)

53. Zemel, R., Wu, Y., Swersky, K., Pitassi, T., Dwork, C.: Learning fair representations. In: International conference on machine learning (ICML). pp. 325–333 (2013)
54. Zhang, M., Ren, Z., Wang, Z., Ren, P., Chen, Z., Hu, P., Zhang, Y.: Membership inference attacks against recommender systems. In: Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS). pp. 864–879 (2021)
55. Zhang, S., Yin, H.: Comprehensive privacy analysis on federated recommender system against attribute inference attacks. IEEE Transactions on Knowledge and Data Engineering (TKDE) (2023)
56. Zhu, T., Li, G., Ren, Y., Zhou, W., Xiong, P.: Differential privacy for neighborhood-based collaborative filtering. In: Proceedings of the IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM). pp. 752–759 (2013)