Exploring and Mitigating Gender Bias in Book Recommender Systems with Explicit Feedback

Shrikant Saxena
Indian Institute of Technology Ropar
Shweta Jain (shwetajain@iitrpr.ac.in)
Indian Institute of Technology Ropar

Research Article

Keywords: Recommender System, Gender Bias, Fairness

Posted Date: July 28th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1876910/v1

Exploring and Mitigating Gender Bias in Book Recommender Systems with Explicit Feedback

Shrikant Saxena[1] and Shweta Jain[2*]

[1*]Computer Science and Engineering, Indian Institute of Technology, Ropar, 140001, Punjab, India.
[2]Computer Science and Engineering, Indian Institute of Technology, Ropar, 140001, Punjab, India.

*Corresponding author(s). E-mail(s): shwetajain@iitrpr.ac.in;
Contributing authors: shrikant.saxena.here@gmail.com;

Abstract

Recommender systems are indispensable because they influence our day-to-day behavior and decisions by giving us personalized suggestions. Services like Kindle, Youtube, and Netflix depend heavily on the performance of their recommender systems to ensure that their users have a good experience and to increase revenues. Despite their popularity, it has been shown that recommender systems reproduce and amplify the bias present in the real world. The resulting feedback creates a self-perpetuating loop that deteriorates the user experience and results in homogenizing recommendations over time. Further, biased recommendations can also reinforce stereotypes based on gender or ethnicity, thus reinforcing the filter bubbles that we live in. In this paper, we address the problem of gender bias in recommender systems with explicit feedback. We propose a model to quantify the gender bias present in book rating datasets and in the recommendations produced by the recommender systems. Our main contribution is to provide a principled approach to mitigate the bias being produced in the recommendations. We theoretically show that the proposed approach provides unbiased recommendations despite biased data. Through empirical evaluation on publicly available book rating datasets, we further show that the proposed model can significantly reduce bias without significant impact on accuracy. Our method is model agnostic and can be applied to any recommender system. To demonstrate the performance of our model, we present the results on four recommender algorithms, two from the K-nearest neighbors family, UserKNN and ItemKNN, and the other two from the matrix factorization family, Alternating least square and Singular value decomposition. The extensive simulations on various recommender algorithms show the generality of the proposed approach.

Keywords: Recommender System, Gender Bias, Fairness

1 Introduction

Recommender systems influence a significant portion of our digital activity. They are responsible for keeping the user experience afresh by recommending varied items from a catalog of millions of items and also adapt their recommendations according to the personality and taste of the user. Therefore, a sound recommender system may go a long way in improving user experience quality, hence the user retentivity of a digital outlet.

Recommender systems have historically been judged on their accuracy (Herlocker et al, 2004; Shani and Gunawardana, 2011). When it is concerned with other factors such as novelty, user satisfaction, and diversity (Hurley and Zhang, 2011; Ziegler et al, 2005a; Knijnenburg et al, 2012), the focus continues to be just on the satisfaction of the information needs of the users. Although of immense importance to the relevance of a recommender system, these criteria do not capture the complete picture. In recent years, the public and academic community have scrutinized artificial intelligence systems regarding their fairness. It has been observed that the results generated by various recommender systems reflect the social biases that exist in human stratum (Ekstrand et al, 2018; Shakespeare et al, 2020; Boratto et al, 2019). Scholars have focused on identifying, quantifying, and mitigating the bias present in the results generated by recommendation systems. Burke (2017) presents a taxonomy of classes for fair recommendation systems. The author suggests different recommendation settings with fairness requirements such as fairness for only users, fairness for only items, and fairness for both users and items. Our work falls into fairness for only items category where bias is shown by a particular set of users against a specific set of items in the dataset. In particular, we are interested in studying and eliminating users' biasedness against the items associated with a specific gender in recommendation systems.

Bias prevention approaches can be classified according to the phase of the data mining process in which they operate: pre-processing, in-processing, and post-processing methods. Pre-processing methods aim to control distortion of the training set. In particular, they transform the training dataset so that the discriminatory biases contained in the dataset are smoothed, hampering the mining of unfair decision models from the transformed data. In-processing methods modify recommendation algorithms such that the resulting models do not entail unfair decisions by introducing a fairness constraint in the optimization problem. Lastly, post-processing methods act on the extracted data mining model results instead of the training data or algorithm. The method presented in our work is a hybrid of a pre-processing phase and a post-processing phase.

Two prominent studies have focused on gender bias in recommender systems. The work by Shakespeare et al (2020) establishes the existence of bias in the results of the music recommender systems, and the work by Ekstrand et al (2018) focuses on bias shown by Collaborative Filtering (CF) algorithms while recommending books written by women authors. Both the studies establish that the CF algorithms produced biased results after being fed the biased data from various socio-cultural factors. While both the works focus just on showing the existence of bias in the presence of the users' implicit feedback, we also consider the explicit feedback ratings and the bias that may arise out of it. Thus, our model handles the case when the items associated with specific gender might have received worse feedback than they otherwise ought to achieve by a set of users. We go one step further and propose a model to mitigate these biases by quantifying a particular user's bias and debiasing his or her feedback ratings. We theoretically show that the debiased ratings are unbiased estimators of the true preference of the user. Once the ratings are debiased, they are fed into the recommender algorithms as input to produce recommendations for the desired set of users. Since the recommender system is now fed with the debiased ratings, the resulting recommendations are free from the bias factor and avoid a self-perpetuating loop in the future.

The bias of an individual user reflects his or her taste. However, the KNN based algorithms produce recommendations based on similar characteristics between a set of users and naive implementation of these algorithms reflects the bias of one user in the recommendations produced for the other user. While not directly comparing the rating history of different users or items, Matrix Factorization algorithms rely on deriving latent factors, which depend on the rating history. Both the approaches make the system increasingly biased and homogenized after users interact with their biased recommendations and generate data for the next iteration. The above discussion suggests that though it is necessary to reflect the user's preference in the recommendations produced for him or her to achieve accuracy, it is equally necessary to prevent the bias of one user from reflecting in the recommendations of another similar user. Our research focuses on this particular objective.

Our debiased ratings assure that the biases of one user do not affect other users; however, it may lead to loss of accuracy because of not reflecting the user's own preferences. We introduce a new step called preference correction which injects the user's preference parameter into his/her own debiased recommendation to maintain the accuracy of the system. The novelty of our work lies in computing the user's preference parameter which not only helps in debiasing the ratings but also in maintaining the preferences of users. On the publicly available Book-Crossing dataset (Ziegler et al, 2005b) and Amazon Book Review dataset (Ni et al, 2019), we empirically show that this approach retained the significant reduction in bias and had minimal effect on the accuracy of the system. The bias reflected in the recommendations produced by the UserKNN, ItemKNN, ALS, and SVD algorithms is reduced by as much as 42.39%, 37.65%, 26.51%, and 41.43% respectively for the Amazon dataset and by 37.82%, 30.73%, 24.99%, and 32.34% for the Book-Crossing dataset. When measured with respect to Root Mean Squared Error(RMSE), the final accuracy loss in the case of the Amazon dataset comes out to be 7.8%, 11.96%, 12.49%, and 10.38% respectively for the four algorithms. In the case of the Book-Crossing dataset, the RMSE loss comes out to be 13.86%, 18.13%, 11.41%, and 12.89% respectively. In particular, the following are our main contributions.

1.1 Contributions

- We propose a model to quantify the gender bias in the recommender system when explicit feedback is present.
- We propose a principled approach to debias the ratings given and theoretically show that the debiased ratings represent the unbiased estimator of the true preference of the user.
- We empirically evaluate our model on publicly available book datasets and show that the approach significantly reduced the biasedness in the system. To show the generality of our proposed approach, we show the results on four algorithms, UserKNN, ItemKNN, ALS, and SVD.
- In order to further enhance the accuracy of the debiased system, we propose an approach of preference correction that respects the user's own preferences towards his/her recommendations. We show that the final recommender system significantly reduces the bias in the system while not deteriorating the accuracy much.

2 Related Works

The problem of gender bias and discrimination has received lots of attention in recent works (Hajian et al, 2016). Many proposals like Pedreschi et al (2008), Pedreschi et al (2009), Ruggieri et al (2010), Thanh et al (2011), Mancuhan and Clifton (2014), Ruggieri et al (2014) are dedicated to detecting and measuring the existing biases in the datasets while other efforts (Kamiran et al, 2010, 2012; Hajian and Domingo-Ferrer, 2013; Hajian et al, 2014a,b; Dwork et al, 2011; Zemel et al, 2013) are focused on ensuring that data mining models do not produce discriminatory results even though the input data may be biased. Most of these works focus on the classical problem of classification. Amatriain et al (2011) discuss the application of various classification methods like Support Vector Machines, Artificial Neural Networks, Bayesian classifiers, and decision trees in recommender systems. Their findings indicated that a more complex classifier need not give a better performance for recommender systems, and more exploration is needed in this direction.

When considering "fairness for only users" according to the taxonomy presented by Burke (2017), Boratto et al (2019) and Tsintzou et al (2018) discuss the bias with respect to the preferential recommending of certain items only to the users of a specific gender. While weighted regularization matrix factorization studied in Boratto et al (2019) is only appropriate for implicit feedback, the Group Utility Loss Minimization proposed in Tsintzou et al (2018)

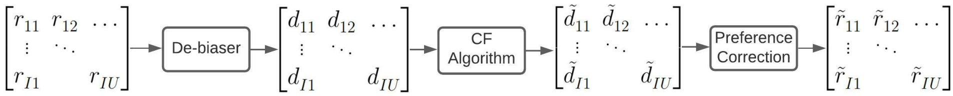


Fig. 1: Model schematics

works only with respect to the UserKNN algorithm. Both the papers address the issue of gender bias by employing post-processing algorithms that work only in limited settings. Though Boratto et al (2019) and Tsintzou et al (2018) have addressed the issue of fairness of recommender systems with respect to gender, they have done so from the perspective of recommending certain items only to the users of a specific gender. The difference between their work and our study lies in the fact that we focus on the more direct issue of gender bias in recommendations shown to items associated with a specific gender.

Shakespeare et al (2020) in their research highlight the artist gender bias in music recommendations produced by Collaborative Filtering algorithms. The work traces the causes of disparity to variations in input gender distributions and user-item preferences, highlighting the effect such configurations can have on user's gender bias after recommendation generation. Mansoury et al (2020) discuss the biases from the perspective of a specific group of individuals (for example, a particular gender) receiving less calibrated and hence unfair recommendations. Ekstrand et al (2018) explores the gender bias present in the book rating dataset. Our work is different from the works by Shakespeare et al (2020), Mansoury et al (2020) and Ekstrand et al (2018) in primarily two factors: (i) we consider explicit feedback as opposed to the implicit feedback, and (ii) we propose a principled approach to debias the ratings and theoretically show that the debiased ratings are unbiased estimators of true ratings.

The research by Leavy et al (2020) focuses on algorithmic gender bias and proposes a framework whereby language-based data may be systematically evaluated to assess levels of gender bias prevalent in training data for machine learning systems. Our work is different from this study as this study is focused on evaluating gender bias in the language and textual data settings, while ours deals with gender bias in a more traditional user-item rating setting.

A couple of works in fair recommender systems focus on improving the exposure of the items belonging to minority groups. They do so by upsampling the items associated with minority groups (Boratto et al, 2021), or by adding more data points to the dataset so as to achieve overall fairness (Rastegarpanah et al, 2019). On the contrary, our goal in this paper is to provide a systematic way to reduce the bias of one user affecting the recommendations to users. We do so via feeding unbiased ratings of the users to the recommender system. This direction avoids the self-perpetuating loop in the recommender system. Once such a system is deployed, there is no further need for interference by the system to ensure fairness. Further, no existing approaches provide a theoretical framework to mitigate the gender bias from the recommender system. We believe this is a strong first step in a new direction for a fair recommender system.

3 The Model

Consider a recommender system having $\mathcal{U} = \{1, 2, \ldots, U\}$ users and $\mathcal{I} = \{1, 2, \ldots, I\}$ items. Let $\mathbb{D}$ and $\mathbb{A}$ denote the set of items associated with disadvantaged group and advantaged group, respectively. For example, in a book recommender system, the books represent the items; $\mathbb{D}$ and $\mathbb{A}$ represent the set of books written by women and men authors respectively. With respect to book recommender system, researchers have already shown that the data is biased against female authors' books (Ekstrand et al, 2018).

Let $r_{ui} \in [1, R]$ denote the rating that user u has given to the item i. As opposed to previous works, we consider explicit feedback wherein biases may not only arise from not giving the rating to the item but may also come from giving a bad rating to the item. The user profile $p_u = \{X_u, R_u\}$ represents the set of books (X_u) and the ratings ($R_u = \{r_{ui}\}_{i \in X_u}$) that user u has given to those items.

The proposed recommender system first pre-processes the data that: 1) finds the log-bias θ_u of each user u and 2) generates the debiased rating d_{ui} of each user u and item i using the computed bias in the first step. We then theoretically show that the debiased ratings generated are unbiased estimators of the true preferences of the user for the items rated by them. Thus, the debiased dataset can then be fed into various recommender algorithms to generate an unbiased predicted rating of a user u for the item i, denoted by $\tilde{d}_{ui}$. This debiasing step ensures that the existing biases are not boosted further in the system. Our debiasing model is independent of any recommendation algorithm. We show the performance of our debiasing model on both K-nearest neighbors-based algorithms (UserKNN, ItemKNN) as well as matrix factorization-based algorithms (Alternating Least Square and Singular Value Decomposition) to produce the recommendations.

In the next step, we use preference corrector to reintroduce the preferences of a particular user u to his/her own recommendations. This is achieved via producing a user specific rating $\tilde{r}_{ui}$ from the debiased rating $\tilde{d}_{ui}$. The recommendations are re-ranked according to the adjusted ratings, and the recommendations are presented to the user. This step ensures that the system does not lose accuracy for not considering the preferences of the users. Figure 1 shows the schematic diagram of our model. Consider that the ratings r_{ui} are continuous values ranging from 1 to R, then mathematically, a biased recommender system can be represented as follows:

1. Each user u, while rating an item i, scales down the maximum rating R by $e^{p_{ui}}$. p_{ui} is a random variable, drawn from a distribution function $P_u(I)$, which has a mean value of α_u. p_{ui} represents the logarithm of the true preference of the user u for the item i. For the sake of brevity, we call it

log-preference of the user u for the item i. Hence $e^{p_{ui}}$ is a representation of the true preference of user u for the item i.

2. In case the item is associated with the disadvantaged group, the user u further scales down the rating of the item by a factor $e^{q_{ui}}$. q_{ui} is a random variable, drawn from a distribution function $Q_u(I)$ having a mean value of β_u. q_{ui} represents the logarithm of the biasedness of the user u shown to the item i. For the sake of brevity, we call it the log-bias of the user u for the book i. Hence $e^{q_{ui}}$ represents the biasedness of the user u for the book i.
3. For each user u, β_u is sampled from the a distribution function $\Omega(x)$ which governs the global log-bias tendency of the users. We denote the mean value of $\Omega(x)$ by γ.

Thus, ratings r_{ui} can be expressed as:

$$r_{ui} = \begin{cases} R/e^{p_{ui}}, & \textit{if } i \textit{ is associated with advantaged group} \\ R/e^{p_{ui}}e^{q_{ui}}, & \textit{if } i \textit{ is associated with disadvantaged group} \end{cases} \tag{1}$$

We now present a detailed description of each of the step.

3.1 Estimating the mean value for log-bias

The geometric mean of the ratings given by a user u to the items associated with disadvantaged and advantaged groups, denoted by r_{ud} and r_{ua} respectively, are given by the following expressions:

$$r_{ud} = \left(\prod_{i \in \mathbb{D} \cap X_u} r_{ui}\right)^{1/|\mathbb{D} \cap X_u|} \quad \text{and} \quad r_{ua} = \left(\prod_{i \in \mathbb{A} \cap X_u} r_{ui}\right)^{1/|\mathbb{A} \cap X_u|}$$

Further, the log bias in the user profile p_u, is given by $\theta_u = \ln\left(\frac{r_{ua}}{r_{ud}}\right)$.

We use geometric mean to compute the average rating of a user due to the following reasons: 1) It is less biased towards very high scores as compared to arithmetic mean (Neve and Palomares, 2019) and 2) when cold users are involved, aggregating recommendations using the geometric mean is more robust as compared to arithmetic mean (Valcarce et al, 2020).

The below lemma shows that θ_u is an unbiased estimator of β_u.

Lemma 1 *The expectation of log-bias, θ_u in the user profile p_u represents the mean value of the log-bias, β_u of the user u.*

Proof Let us denote $m = |\mathbb{D} \cap X_u|$ and $n = |\mathbb{A} \cap X_u|$ to be the number of items associated with disadvantaged and advantaged group respectively in user profile p_u.

Then,

$$\theta_u = \ln\left(\frac{r_{ua}}{r_{ud}}\right) = \ln\left[\frac{\left(\prod_{y=1}^{m} e^{p_{uy}} e^{q_{uy}}\right)^{\frac{1}{m}}}{\left(\prod_{x=1}^{n} e^{p_{ux}}\right)^{\frac{1}{n}}}\right] \qquad \text{(Using equation 1)}$$
$$= \frac{1}{m}\sum_{y=1}^{m} q_{uy} + \frac{1}{m}\sum_{y=1}^{m} p_{uy} - \frac{1}{n}\sum_{x=1}^{n} p_{ux}$$

Taking expectation both sides:

$$\mathbb{E}[\theta_u] = \mathbb{E}\left[\frac{1}{m}\sum_{y=1}^{m} q_{uy} + \frac{1}{m}\sum_{y=1}^{m} p_{uy} - \frac{1}{n}\sum_{x=1}^{n} p_{ux}\right] \tag{2}$$

Using linearity of expectation and some simplification, we get:

$$\mathbb{E}[\theta_u] = \frac{1}{m}\sum_{y=1}^{m} \mathbb{E}[q_{uy}] + \frac{1}{m}\sum_{y=1}^{m} \mathbb{E}[p_{uy}] - \frac{1}{n}\sum_{x=1}^{n} \mathbb{E}[p_{ux}]$$
$$= \frac{1}{m}\sum_{y=1}^{m} \beta_u + \frac{1}{m}\sum_{y=1}^{m} \alpha_u - \frac{1}{n}\sum_{x=1}^{n} \alpha_u$$

Thus, $\mathbb{E}[\theta_u] = \beta_u$. □

Once we get the log biasedness tendencies of users, we use them to produce the debiased ratings for the given dataset.

3.2 Debiasing the Dataset

The debiased rating of the item i associated with disadvantaged group and rated by user u is given as $d_{ui} = r_{ui}e^{\theta_u}$ We now provide the main theorem of our paper.

Theorem 2 $\ln(d_{ui})$ *is the unbiased estimator of the log of the true rating of the item* i*.*

Proof $\ln(d_{ui}) = \theta_u + \ln(r_{ui}) = \theta_u + \ln R - p_{ui} - q_{ui}$. Last equality is obtained from Equation 1. Taking expectation both sides:

$$\mathbb{E}(\ln(d_{ui})) = \mathbb{E}[\theta_u] + \mathbb{E}[\ln R] - \mathbb{E}[q_{ui}] - \mathbb{E}[p_{ui}]$$
$$= \beta_u + \ln R - \beta_u - \alpha_u \qquad \text{(Using Lemma 1)}$$
$$= \ln R - \alpha_u = \ln\left(\frac{R}{e^{\alpha_u}}\right)$$

As we can see, the expected value of $\ln(d_{ui})$ contains only the term representing the true preference of the item for user u. □

Thus, instead of r_{ui}, ratings d_{ui} are fed into the recommender system to generate the predicted unbiased ratings $\tilde{d}_{ui}$. Simply removing the bias from the user's rating could severely affect the system's accuracy because the bias of an individual user reflects their taste. However, the debiasing step helps prevent the bias of one user from affecting the recommendation of other users. Next, we use preference corrections by correcting the predicted rating of the user with respect to his/her own preference parameter.

3.3 Preference Correction to Improve Accuracy

Note that when the users are inherently biased against a group of items, $\mathcal{D}$ then showing the items from $\mathcal{D}$ naively to these users will severely affect the accuracy of the system. The goal of this work is not just to promote the exposure of the items among the two groups but is to not let the bias of one user creep into the bias of the other user. This was achieved via debiasing the dataset. Once the debiased ratings are generated, the accuracy of the system is maintained by introducing a correction factor. Although providing us with higher accuracy, the idea to re-introduce the correction factor may lead to an overall increase in the individual biases. This on a prima-facie may look self-defeating, but we need to note that final ratings still have significantly less bias than original ratings. If we do not introduce the correction factor, the users might flock to a substantial bias platform due to poor accuracy.

The correction is achieved via multiplying the predicted ratings of items associated with disadvantaged group by a factor $e^{-\theta_u}$. Thus, the final recommended ratings will be given as $\tilde{r}_{ui} = \tilde{d}_{ui}e^{-\theta_u}$. Similar to the calculation of bias in the dataset, we can now compute the bias in the recommendation profile.

3.4 Bias in recommendation profile

We generate recommendations for the users in the test set $\mathcal{T}$. The recommendation profile for a user $u \in \mathcal{T}$ is denoted by $\tilde{p}_u = \{\tilde{X}_u, \tilde{R}_u\}$, which represents the set of recommended books $(\tilde{X}_u)$ for the user u and their predicted ratings $(\tilde{R}_u = \{\tilde{r}_{ui}\}_{i\in\tilde{X}_u})$. Let the set of items associated with disadvantaged and advantaged groups be denoted by $\tilde{\mathbb{D}}$ and $\tilde{\mathbb{A}}$ respectively. The average predicted ratings of the items associated with disadvantaged and advantaged groups, denoted by $\tilde{r}_{ud}$ and $\tilde{r}_{ua}$ respectively, are given by: $\tilde{r}_{ud} = \left(\prod_{i\in\tilde{\mathbb{D}}\cap\tilde{X}_u} \tilde{r}_{ui}\right)^{1/|\tilde{\mathbb{D}}\cap\tilde{X}_u|}$ and $\tilde{r}_{ua} = \left(\prod_{i\in\tilde{\mathbb{A}}\cap\tilde{X}_u} \tilde{r}_{ui}\right)^{1/|\tilde{\mathbb{A}}\cap\tilde{X}_u|}$ where $\tilde{r}_{ui}$ is the predicted rating given to item i in the recommendation-profile generated for a user u. The log-bias in the recommendation-profile p_u, denoted by $\tilde{\theta}_u$, is then given by $\tilde{\theta}_u = \ln\left(\frac{\tilde{r}_{ua}}{\tilde{r}_{ud}}\right)$. For an unbiased recommendation-profile, $\tilde{\theta}_u = 0$. A profile biased against disadvantaged groups will have $\tilde{\theta}_u > 0$. We can then compute the overall bias of the recommender system by taking the average overall users, and this average gives us the estimated value of γ.

4 Dataset

To evaluate the proposed model, we run experiments on two publicly available book rating datasets, the Book-Crossing dataset, originally put together by

Statistic	Amazon	Book-Crossing
Number of male authored books	58369	829
Number of female authored books	58220	806
Number of users	44792	376

Table 1: Dataset details

Ziegler et al (2005b) and the Amazon Book Review dataset, put together by Ni et al (2019). We further process this dataset through the following stages:

4.1 Book Author Identification

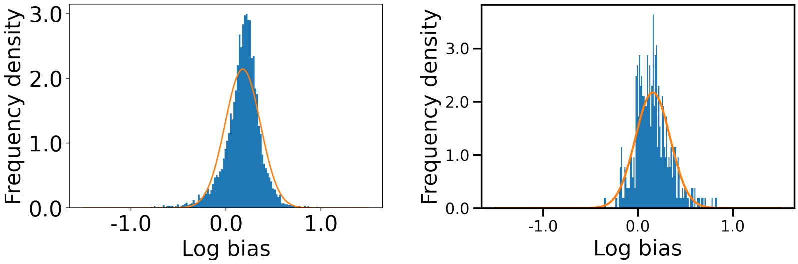
Their unique ISBNs identify the books in both datasets. We identified the authors of the books present in the datasets via their ISBN numbers using the following three API services: *Google Books API* APIs (Accessed: 2021-02-24), *ISBNdb API* ISBNDB (Accessed: 2021-02-27), and *Open Library API* OpenLibrary (Accessed: 2021-03-02). We could not identify the authors of some of the books. Hence we discarded those books from the dataset.

4.2 Author Gender Identification

We identified the genders of the authors via their first names. We used *Genderize.io* the gender of a name (Accessed: 2021-03-5), an API service dedicated to identifying the gender given the first name of the person. We used a minimum confidence threshold of 90% for gender identification. We could not identify the gender of some of the authors. We discard the books written by those authors from the dataset.

4.3 Filtering

We filtered the Book-Crossing dataset to include only those books with at least 50 ratings and only those users who have rated at least 50 books. Amazon dataset was significantly larger as compared to the Book-Crossing dataset. We filtered it to include only those books with at least 100 ratings and only those users who have rated at least 100 books. We did this filtering so that recommender algorithms have much data to produce accurate recommendations. The statistics of filtered datasets are mentioned in Table 1. The number of books written by male authors is almost equal to that of female authors for both datasets.



(a) Amazon dataset

(b) Book-Crossing dataset

Fig. 2: User log-bias in the original dataset

5 Experimental Results

5.1 Input Bias

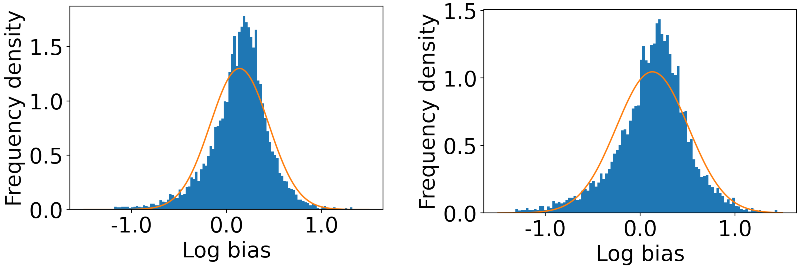
We show the distributions of log-bias tendency (θ_u) of the users in the Amazon dataset and the Book-Crossing dataset in Figure 2. We observe that the mean log-bias tendency over all the users in the Amazon dataset is higher (0.176) than that of the Book-Crossing dataset (0.157)[1] .

5.2 Output Bias

We randomly separate 20% of users in each dataset as the test group. We generate the recommendations for the users in the test group using two K-nearest neighbors-based algorithms, UserKNN and ItemKNN, and two matrix factorization-based algorithms, Alternating Least Square and Singular Value Decomposition. These algorithms were selected because the accuracy and ranking relevancy of the recommendations produced by them were among the highest values compared with other algorithms. Hence coupling our model with them would best highlight the effects brought about by the same. We calculate the estimated value of log-bias ($\tilde{\theta}_u$) and accuracy in the recommendations separately for each algorithm applied on the two datasets. For this, we use two error measures, the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE), and two ranking relevance parameters, Normalized Discounted Cumulative Gains and Mean Reciprocal Rank.

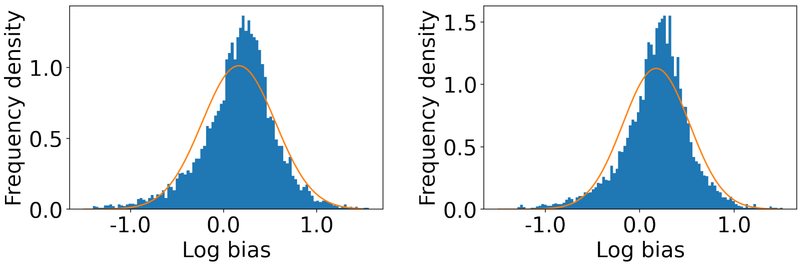
We first begin plotting the log-bias ($\tilde{\theta}_u$) distribution for the recommendations produced by the algorithms without employing our debiased model in Figures 3 and 4 for Amazon datasets with respect to K-nearest neighbor family and matrix factorization family of algorithms. Figures 5 and 6 similary present the log-bias distribution for the recommendations produced by the two family of algorithms for Book-Crossing datasets respectively without employing our debiased model. We compute the log-bias by feeding biased ratings r_{ui} to the

[1]code is available at https://github.com/venomNomNom/genderBias.git



(a) UserKNN (b) ItemKNN

Fig. 3: Output log-bias in AZ dataset without employing the model under K-nearest neighbour family of algorithms



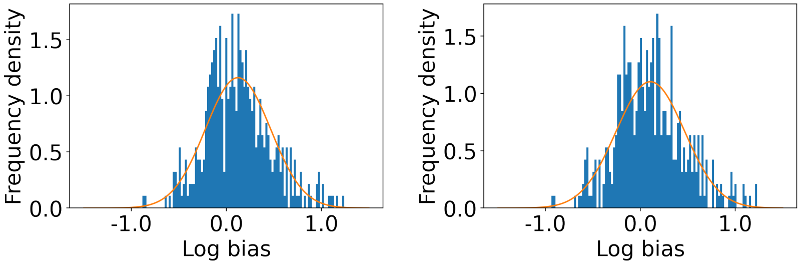
(a) ALS (b) SVD

Fig. 4: Output log-bias in AZ dataset without employing the model under matrix factorization family of algorithms

four algorithms. As can be seen from the figures, that the output log biasedness was very similar to what was observed in the input data.

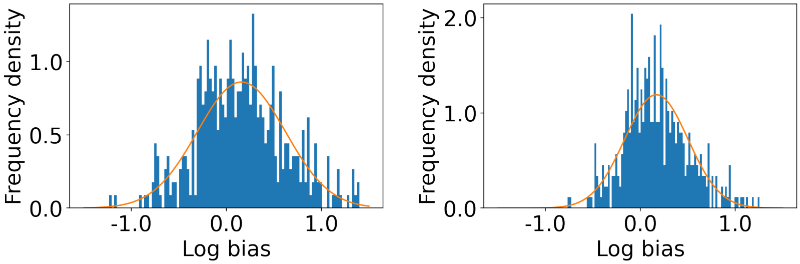
We next deploy our model partially. We leave out the preference correction phase and produce the recommendations using the algorithms mentioned before by feeding the debiased ratings d_{ui} to these algorithms. We estimate the mean log-bias tendency in the recommendations $\tilde{\theta}_u$ using debiased ratings produced by the algorithms $\tilde{d}_{ui}$. The log-bias $(\tilde{\theta_u})$ distribution for the recommendations produced by the algorithms after partial deployment of the model is depicted in the Figures 7 and 8 for Amazon dataset and in the Figures 9 and 10 for book crossing dataset. As can be seen, there is a significant reduction in log-bias tendency (64.38%) in the Amazon dataset and (53.67%) in Book-Crossing dataset for the UserKNN algorithm. However, we also see an increase in error rates on both datasets. This is because the test data itself contains biases.

Finally, we deploy our complete model after adding the preference correction method and repeat the experiment. The log-bias $(\tilde{\theta_u})$ distribution for the recommendations produced by the algorithms after deployment of the complete



(a) UserKNN (b) ItemKNN

Fig. 5: Output log-bias in BX dataset without employing the model under K-nearest neighbour family of algorithms



(a) ALS (b) SVD

Fig. 6: Output log-bias in BX dataset without employing the model under matrix factorization family of algorithms

model is depicted in Figures 11, 12 for Amazon dataset and in Figures 13, 14 for book crossing dataset. The final values for all the cases are given in Table 2 for Amzaon dataset and in Table 3 for book crossing datasets. As can be seen, there is still a significant reduction in mean log-bias tendency, which reduces by 42.39% in the Amazon dataset and by 37.82% in the case of the Book-Crossing dataset for UserKNN algorithm. The accuracy loss, however, is insignificant, making this trade-off advantageous. Figure 15 presents the percentage gain in bias reduction for both the dataset. The percentage loss in accuracy is depicted in figures 16 and 17 for Amazon and Book-Crossing datasets respectively. The percentage loss in ranking relevancy metrics are depicted in figures 18 and 19 respectively.

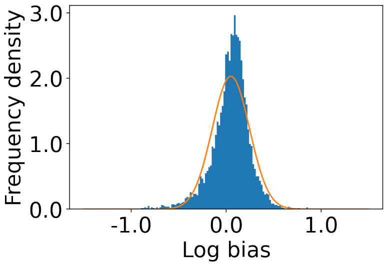
We next conduct significance testing to validate the log-bias reduction. Tables 4 and 5 show the p-values obtained from left-tail significance tests on the log-bias of the recommendations made for the users in the sample. We can see from the p-value for the Amazon datasets that the bias reduction is significant. For the Book-Crossing dataset, the significance of the bias reduction is less pronounced. One of the prominent reasons for this is that the test sample size

Case	Algorithm	Mean log-bias	RMSE	MAE	NDCG	Ranking Relevancy
without model	UserKNN	0.137	0.808	0.693	0.452	0.498
	ItemKNN	0.129	0.736	0.580	0.597	0.643
	ALS	0.164	0.873	0.829	0.281	0.447
	SVD	0.175	0.790	0.753	0.342	0.471
without preference correction phase	UserKNN	0.049	1.103	0.921	0.0224	0.0278
	ItemKNN	0.063	1.076	0.873	0.0229	0.0204
	ALS	0.093	1.281	1.1211	0.0161	0.0394
	SVD	0.071	1.257	1.183	0.0138	0.0206
with preference correction phase	UserKNN	0.079	0.871	0.738	0.3982	0.4462
	ItemKNN	0.080	0.824	0.661	0.5236	0.6121
	ALS	0.121	0.982	0.903	0.2391	0.3853
	SVD	0.103	0.872	0.847	0.2989	0.4159

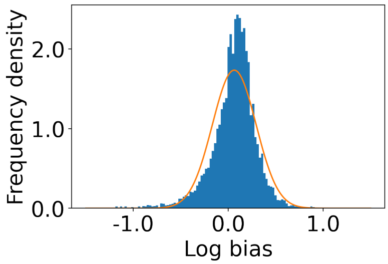
Table 2: Summary of Results for Amazon Dataset

Case	Algorithm	Mean log-bias	RMSE	MAE	NDCG	Ranking Relevancy
without model	UserKNN	0.122	1.580	1.178	0.264	0.272
	ItemKNN	0.106	1.511	1.304	0.313	0.412
	ALS	0.158	1.815	1.642	0.235	0.370
	SVD	0.169	1.761	1.626	0.277	0.296
without preference correction phase	UserKNN	0.057	2.468	1.754	0.0232	0.0245
	ItemKNN	0.054	2.463	2.055	0.0142	0.0271
	ALS	0.087	2.752	2.175	0.0421	0.0736
	SVD	0.072	2.601	1.979	0.0261	0.0240
with preference correction phase	UserKNN	0.076	1.799	1.298	0.2099	0.2317
	ItemKNN	0.073	1.785	1.516	0.2538	0.3560
	ALS	0.119	2.022	1.768	0.1626	0.2831
	SVD	0.114	1.988	1.731	0.2258	0.2481

Table 3: Summary of Results for Bookcrossing Dataset

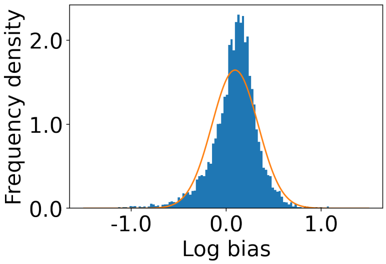


(a) UserKNN

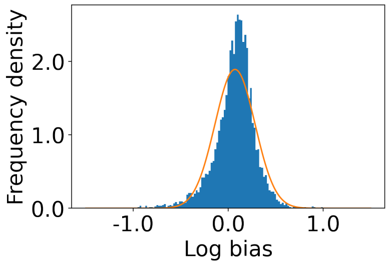


(b) ItemKNN

Fig. 7: Output log-bias in AZ dataset with debiasing under family of K-nearest neighbour algorithms



(a) ALS



(b) SVD

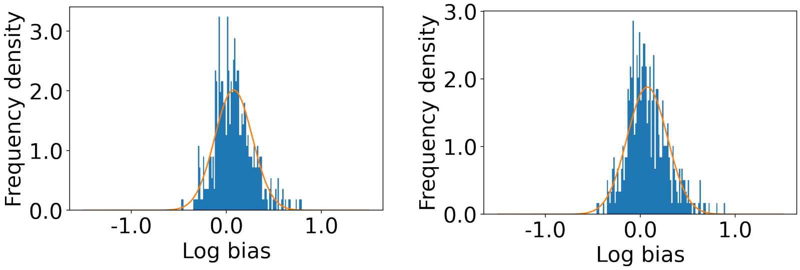
Fig. 8: Output log-bias in AZ dataset with debiasing under family of matrix factorization algorithms

Algorithm	$\bar{x}$	μ	σ	z	p
UserKNN	0.079	0.137	0.307	-17.90	$< 10^{-5}$
ItemKNN	0.080	0.129	0.381	-12.06	$< 10^{-5}$
ALS	0.121	0.164	0.394	-10.46	$< 10^{-5}$
SVD	0.103	0.175	0.354	-19.27	$< 10^{-5}$

Table 4: Significance test results for bias reduction for Amazon Dataset

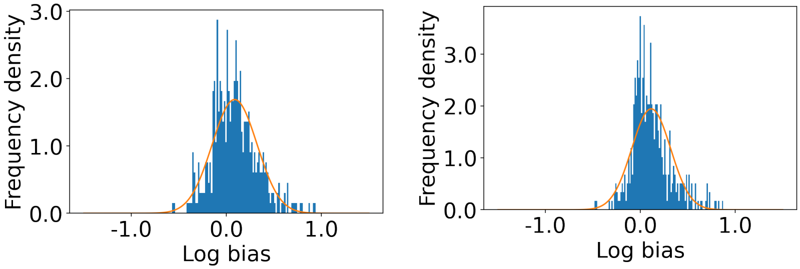
for the Book-Crossing dataset was relatively small due to the small number of users in the dataset. In essence, the utility of the recommender system is maintained while reducing the log-bias tendency in the recommendations.

We further observe that the bias reduction is more in the case of UserKNN based recommendations than the ItemKNN based recommendations. This observation can be attributed to the fact that our model addresses the bias originating from the distortion in ratings from the users' side. It compares the ratings of an item given by a particular user with the appropriately scaled average of ratings given by other users to that item in the dataset. It, therefore,



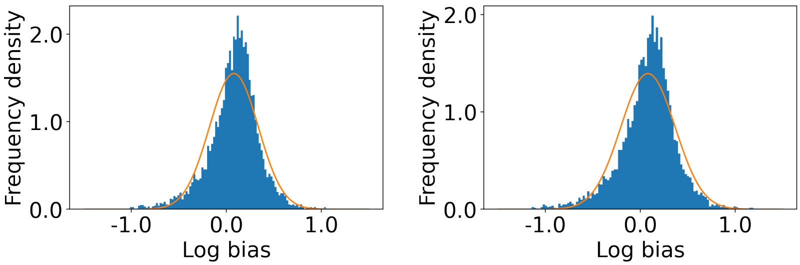
(a) UserKNN (b) ItemKNN

Fig. 9: Output log-bias in BX dataset with debiasing under family of K-nearest neighbour algorithms



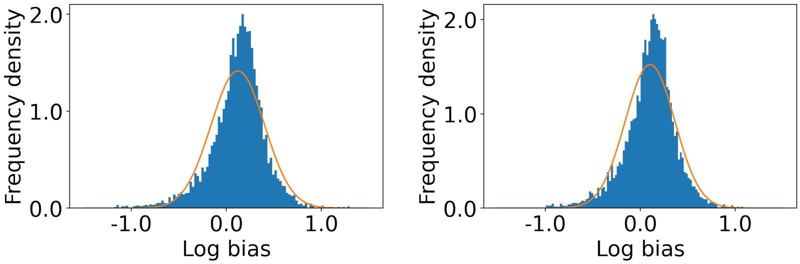
(a) ALS (b) SVD

Fig. 10: Output log-bias in BX dataset with debiasing under family of matrix factorization algorithms



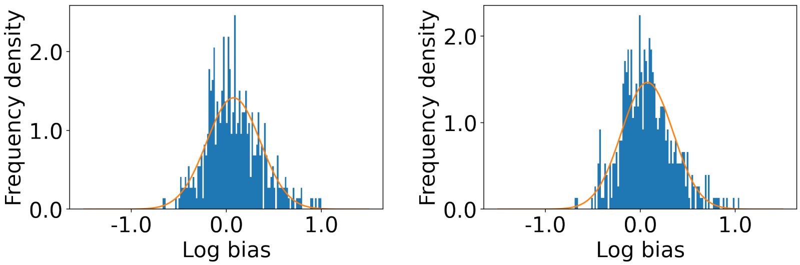
(a) UserKNN (b) ItemKNN

Fig. 11: Output log-bias in AZ dataset with Preference correction under family of K-nearest neighbour algorithms



(a) ALS (b) SVD

Fig. 12: Output log-bias in AZ dataset with Preference correction under family of matrix factorization algorithms



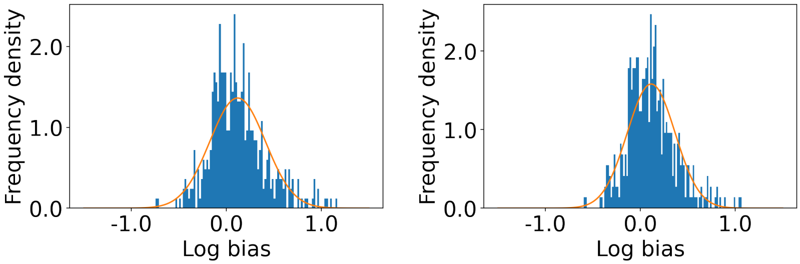
(a) UserKNN (b) ItemKNN

Fig. 13: Output log-bias in BX dataset with reinserting the biases under family of K-nearest neighbour algorithms

Algorithm	$\bar{x}$	μ	σ	z	p
UserKNN	0.076	0.122	0.343	-1.164	0.122
ItemKNN	0.073	0.106	0.362	-0.780	0.218
ALS	0.119	0.158	0.464	-0.738	0.230
SVD	0.114	0.169	0.335	-1.413	0.079

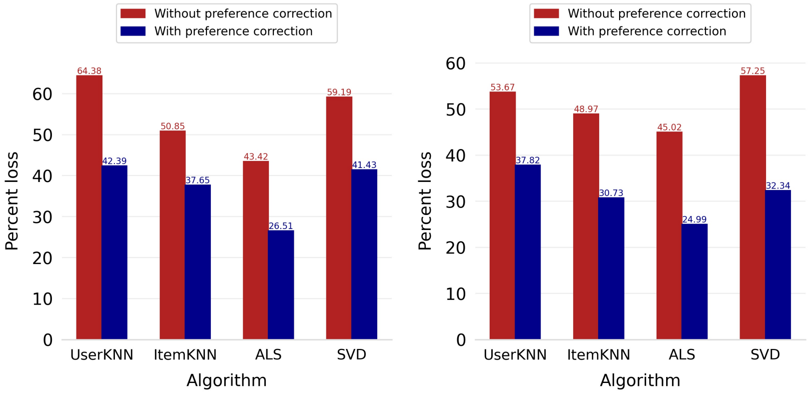
Table 5: Significance test results for bias reduction for Bookcrossing Dataset

resonates with the UserKNN algorithm, which predicts the ratings of an item for a particular user based on the ratings of that item for his or her peers. The ItemKNN algorithm, on the other hand, predicts the ratings of an item for a particular user based on the ratings given to similar items by that user. The model does not sit squarely with ItemKNN. Thus the bias reduction in UserKNN is more as compared to that in the case of ItemKNN. We further observe that the bias reduction is more in the case of the AZ dataset as compared to the BX dataset. This observation can be attributed to the AZ dataset having a higher input mean log-bias tendency. Further, the AZ dataset has a significantly larger



(a) ALS (b) SVD

Fig. 14: Output log-bias in BX dataset with reinserting the biases under family of matrix factorization algorithms

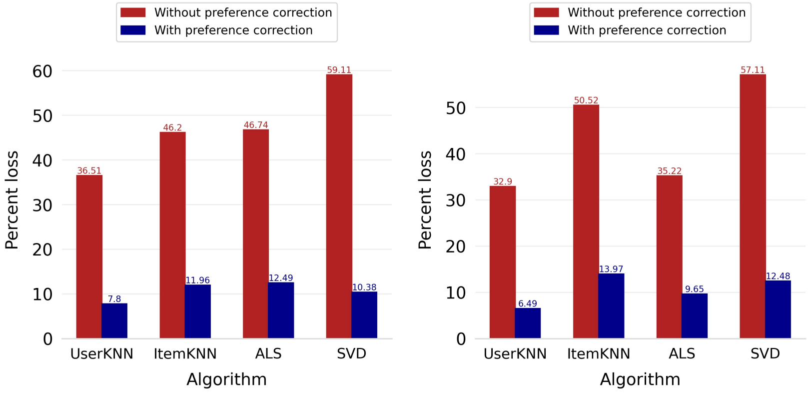


(a) AZ dataset (b) BX dataset

Fig. 15: Bias reduction

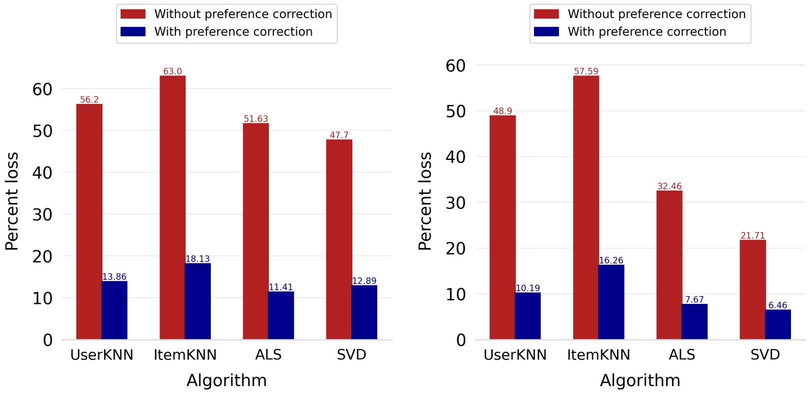
number of users and items which leads to a more accurate estimation of user bias scores and, therefore, more effective bias mitigation.

We observe that accuracy and ranking relevancy loss is, in general, higher for ItemKNN as compared to UserKNN. This is due to the fact that the model quantifies the bias of users by comparing the ratings given by them to particular items with a scaled average of ratings given by their peers to those items. This resonates with the UserKNN algorithm, which predicts user ratings for particular items based on the ratings of similar users. Thus the model is better oriented towards the UserKNN algorithm, giving better accuracy and bias reduction in its case. In the case of matrix factorization algorithms, the accuracy and ranking relevancy losses are relatively comparable. It is not clear which one of the two algorithms is more coherent with the model.



(a) In terms of RMSE (b) In terms of MAE

Fig. 16: Accuracy loss for AZ dataset



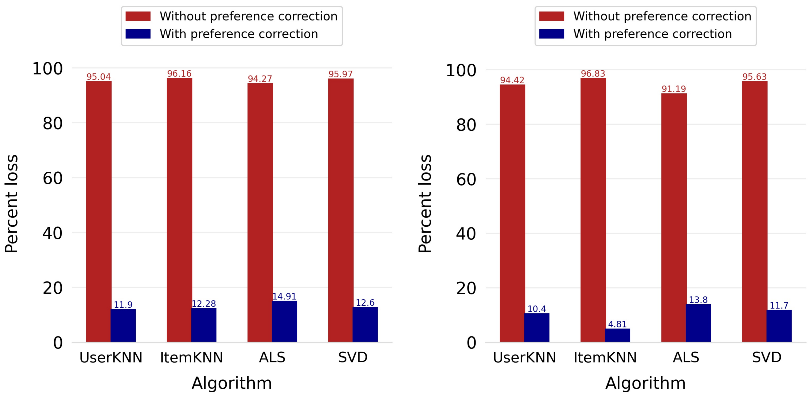
(a) In terms of RMSE (b) In terms of MAE

Fig. 17: Accuracy loss for BX dataset

We further observe that accuracy loss on BX dataset is higher than that of AZ dataset. This observation can be attributed to the fact that the user and item base of the AZ dataset is higher as compared to the BX dataset. Thus, the bias score estimates are more accurate, which provides more accurate predictions of the item scores for the users when reinserted into the recommendations.

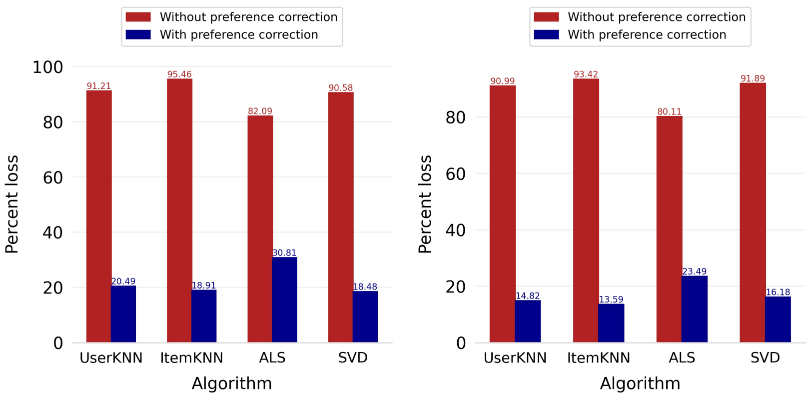
6 Conclusion and Future Work

We proposed a model to quantify and mitigate the bias in the explicit feedback given by the users to different items. We theoretically showed that the debiased



(a) In terms of NDCG (b) In terms of Reciprocal Rank

Fig. 18: Ranking relevancy loss for AZ dataset



(a) In terms of NDCG (b) In terms of Reciprocal Rank

Fig. 19: Ranking relevancy loss for BX dataset

ratings produced by our model are unbiased estimators of the true preference of the users for the books. With the help of comprehensive experiments on two publically available book datasets, we show a significant reduction in the bias (almost 40%) with just 10% decrease in accuracy using the UserKNN algorithm. Similar trends were observed for other algorithms such as ItemKNN, ALS, and SVD. Our model is independent of these algorithms' choices and can be applied with any recommendation algorithm. We used book recommender system because we were able to generate the gender information from publicly available APIs. Our model is not restricted to book recommender system as long as protected attribute information about the items is known. We leave extension of the model to missing protected attribute as an interesting future work. It will be an interesting direction to see if the ideas from fair classification literature with missing protected attributes (Coston et al, 2019) can be leveraged. We further did not address the bias originating from fewer ratings for a female-authored book than a male-authored one. We leave extending the model to the bias originating from lesser number of ratings and extensively studying the model for other recommender systems as the future directions.

Declaration

Ethical Approval and Consent to participate

All authors certify that they have no affiliations with or involvement in any organization or entity with any fnancial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

Consent for publication

Authors give full consent for publications

Human and Animal Ethics

Not Applicable

Availability of supporting data

Dataset is publicly available and code is already made available on github.

Competing interests

The authors declare that they have no confict of interest.

Funding

The project is supported by Science and Engineering Research Board, India with project number SRG/2020/001138.

Authors' contributions

Shrikant Saxena is the main contributing author of the paper. Shweta Jain has helped in writing the manuscript. All authors have reviewed the paper.

Acknowledgments

Not applicable.

References

Amatriain X, Jaimes A, Oliver N, et al (2011) Data Mining Methods for Recommender Systems, pp 39–71. https://doi.org/10.1007/978-0-387-85820-3_2

APIs GB (Accessed: 2021-02-24) URL https://developers.google.com/books

Boratto L, Fenu G, Marras M (2019) The Effect of Algorithmic Bias on Recommender Systems for Massive Open Online Courses, pp 457–472. https://doi.org/10.1007/978-3-030-15712-8_30

Boratto L, Fenu G, Marras M (2021) Interplay between upsampling and regularization for provider fairness in recommender systems. User Modeling and User-Adapted Interaction 31(3):421–455

Burke R (2017) Multisided fairness for recommendation. rXiv preprint arXiv:170700093

Coston A, Ramamurthy KN, Wei D, et al (2019) Fair transfer learning with missing protected attributes. In: Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society, pp 91–98

Dwork C, Hardt M, Pitassi T, et al (2011) Fairness through awareness. CoRR abs/1104.3913. https://doi.org/10.1145/2090236.2090255

Ekstrand M, Tian M, Kazi M, et al (2018) Exploring author gender in book rating and recommendation. User modeling and user-adapted interaction pp 377–420. https://doi.org/10.1145/3240323.3240373

Hajian S, Domingo-Ferrer J (2013) A methodology for direct and indirect discrimination prevention in data mining. IEEE Transactions on Knowledge and Data Engineering https://doi.org/10.1109/TKDE.2012.72

Hajian S, Domingo-Ferrer J, Farràs O (2014a) Generalization-based privacy preservation and discrimination prevention in data publishing and mining. Data Mining and Knowledge Discovery https://doi.org/10.1007/s10618-014-0346-1

Hajian S, Domingo-Ferrer J, Monreale A, et al (2014b) Discrimination- and privacy-aware patterns. Data Mining and Knowledge Discovery 29. https://doi.org/10.1007/s10618-014-0393-7

Hajian S, Bonchi F, Castillo C (2016) Algorithmic bias: From discrimination discovery to fairness-aware data mining. pp 2125–2126, https://doi.org/10.1145/2939672.2945386

Herlocker JL, Konstan JA, Terveen LG, et al (2004) Evaluating collaborative filtering recommender systems. ACM Trans Inf Syst 22(1):5–53. https://doi.

org/10.1145/963770.963772, URL https://doi.org/10.1145/963770.963772

Hurley N, Zhang M (2011) Novelty and diversity in top-n recommendation – analysis and evaluation. ACM Trans Internet Technol 10(4). https://doi.org/10.1145/1944339.1944341, URL https://doi.org/10.1145/1944339.1944341

ISBNDB ID (Accessed: 2021-02-27) URL https://isbndb.com/isbn-database

Kamiran F, Calders T, Pechenizkiy M (2010) Discrimination aware decision tree learning. pp 869–874, https://doi.org/10.1109/ICDM.2010.50

Kamiran F, Karim A, Zhang X (2012) Decision theory for discrimination-aware classification. pp 924–929, https://doi.org/10.1109/ICDM.2012.45

Knijnenburg B, Willemsen M, gantner s, et al (2012) Explaining the user experience of recommender systems. User Modeling and User-Adapted Interaction 22:441–504. https://doi.org/10.1007/s11257-011-9118-4

Leavy S, Meaney G, Wade K, et al (2020) Mitigating gender bias in machine learning data sets. International Workshop on Algorithmic Bias in Search and Recommendation pp 12–26

Mancuhan K, Clifton C (2014) Combating discrimination using bayesian networks. Artificial Intelligence and Law 22. https://doi.org/10.1007/s10506-014-9156-4

Mansoury M, Abdollahpouri H, Smith J, et al (2020) Investigating potential factors associated with gender discrimination in collaborative recommender systems. The Thirty-Third International Flairs Conference

the gender of a name GD (Accessed: 2021-03-5) URL https://genderize.io/

Neve J, Palomares I (2019) Latent factor models and aggregation operators for collaborative filtering in reciprocal recommender systems. In: Proceedings of the 13th ACM Conference on Recommender Systems, pp 219–227

Ni J, Li J, McAuley J (2019) Justifying recommendations using distantly-labeled reviews and fine-grained aspects. pp 188–197, https://doi.org/10.18653/v1/D19-1018

OpenLibrary DA (Accessed: 2021-03-02) URL https://openlibrary.org/developers/api

Pedreschi D, Ruggieri S, Turini F (2008) Discrimination-aware data mining. pp 560–568, https://doi.org/10.1145/1401890.1401959

Pedreschi D, Ruggieri S, Turini F (2009) Measuring discrimination in socially-sensitive decision records. pp 581–592, https://doi.org/10.1137/1.

9781611972795.50

Rastegarpanah B, Gummadi KP, Crovella M (2019) Fighting fire with fire: Using antidote data to improve polarization and fairness of recommender systems. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, pp 231–239

Ruggieri S, Pedreschi D, Turini F (2010) Data mining for discrimination discovery. TKDD 4. https://doi.org/10.1145/1754428.1754432

Ruggieri S, Hajian S, Kamiran F, et al (2014) Anti-discrimination analysis using privacy attack strategies. https://doi.org/10.1007/978-3-662-44851-9_44

Shakespeare D, Porcaro L, Gómez E, et al (2020) Exploring artist gender bias in music recommendation. arXiv preprint arXiv:200901715

Shani G, Gunawardana A (2011) Evaluating Recommendation Systems, vol 12, pp 257–297. https://doi.org/10.1007/978-0-387-85820-3_8

Thanh B, Ruggieri S, Turini F (2011) k-nn as an implementation of situation testing for discrimination discovery and prevention. pp 502–510, https://doi.org/10.1145/2020408.2020488

Tsintzou V, Pitoura E, Tsaparas P (2018) Bias disparity in recommendation systems. arXiv preprint arXiv:181101461

Valcarce D, Bellogín A, Parapar J, et al (2020) Assessing ranking metrics in top-n recommendation. Information Retrieval Journal 23:411–448

Zemel R, Wu Y, Swersky K, et al (2013) Learning fair representations. 30th International Conference on Machine Learning, ICML 2013 pp 1362–1370

Ziegler CN, McNee SM, Konstan JA, et al (2005a) Improving recommendation lists through topic diversification. Association for Computing Machinery, New York, NY, USA, WWW '05, p 22–32, https://doi.org/10.1145/1060745.1060754, URL https://doi.org/10.1145/1060745.1060754

Ziegler CN, McNee SM, Konstan JA, et al (2005b) Improving recommendation lists through topic diversification. In: Proceedings of the 14th International Conference on World Wide Web. Association for Computing Machinery, New York, NY, USA, WWW '05, p 22–32, https://doi.org/10.1145/1060745.1060754, URL https://doi.org/10.1145/1060745.1060754