Computer Science
&
Data Science

Capstone Report - Spring 2022

Improving Fairness in Recommender Systems with Multi-subgroup Debiasing and DCNF Model

Yuxuan Wang,
Xingjian Gao

supervised by
Hongyi Wen

Acknowledgements

Thanks to Professor Hongyi Wen for supporting us through the entire project. Thanks to the previous scholars, including but not limited to the authors mentioned in the reference, for doing such wonderful works which enlightened us in the field of recommender systems. Thank to our parents and loved ones, for supporting us, in this project and in life.

Abstract

While modern recommender system has achieved good overall performance, they often rely significantly on the explicit information, and the fairness of the recommender system across different subgroups cannot be guaranteed. We extended upon previous literature and proposed our own method of multi subgroup debiasing together with the DCNF model that incorporates this method. The debiasing method is proved instrumental in improving fairness in multi subgroups by our visualization of recommended careers. In addition, it does not over-correct the embedding and avoids exerting apparent impact on the fairness of other subgroup measures. The DNCF model has displayed consistency in multiple specifications, and the overall performance and fairness exceeds our baseline models. When the target subgroup matches the debiasing subgroup, the model produces satisfactory results. Our results agree with the literature that regularization serves as a trade-off between performance and fairness. The stronger regularization, the better fairness and worse model performance. In the setting of this study, the increased regularization considerably improves fairness and only mildly impacts performance.

Keywords

Neural Collaborative Filtering; Matrix Factorization; Fairness Metric

Contents

1 Introduction

In recent years, as Internet has become an indispensable component of people's lives, the recommender system, as a presonalization tool, has become more and more vital. The recommender system can collect the behavioural data of the users, for instance, whether the user prefer the pages related to cooking or the pages related to sports, or for movie website, whether the user prefer horror movies or animations. Through collecting the users' personal data, the recommender system can use the corresponding algorithm to predict the users' preference, and thus recommend the items that the users are most likely to be interested in. The recommender system has been widely used in all kinds of areas, in people's everyday lives. Therefore, it is vital to train the recommender systems properly so that it can give a satisfactory performance. Although modern recommender systems can achieve good overall performance, they often rely on explicit user feedback which is sometimes unavailable. The explicit feedback, for instance, the ratings for the movie, is only a small portion of the massive data created by the users. To make an accurate prediction, one also has to take the implicit feedback into consideration, so the collaborative filtering (CF) method is invented. The popular CF method includes the the Matrix Factorization (MF) and Neural Collaborative Filtering (NCF). These models have achieved expressive performances in the recommender systems.

However, another concern for the recommender system is whether it is fair or not. Since for many CF method, the users and items are represented by embedding vectors, the value of the vectors could be influenced by all kinds of data, and bias occurred by certain data could influence the embedding, and thus influence other recommendations. For instance, female users have a higher chance of buying skirts than the male users, so the embedding vectors for female users and male users could be inluenced by this specific interaction, and for more crucial recommendations, like job recommendation. When the user embedding is changed by the bias of skirts, the difference within the embedding for female users and male users will also cause biased performance when recommending the jobs. Thus, the bias from the skirts can indirectly influnce the job. But when a recommendation algorithm only focuses on the average performance but ignored the performance in each subgroup, the bias among subgroups could be exacerbated, and cause a subgroup of users to have a bad experience. To address the unfairness occurred in the recommender system, there have been many works trying to alleviate this issue, and some of them have achieved exciting results. In this project we would emphasize on ensuring the fairness of final ranking across different subgroups of consumers and items. In addition, improved fairness should not be accompanied with degraded model utility. We look deep into the definition of fairness, evaluate different metrics for fairness and choose an appropriate metric for our research. We develop an algorithm based on existing approaches like NCF (Neural Collaborative Filtering) to process implicit feedback of consumers, and attempt to enhance algorithmic fairness in both procedures and especially in the latter.

2 Related Work

The modern recommender system has achieved very satisfying overall performance, it has been used in many areas, for instance, the Youtube and some shopping apps. For its universality, the recommender system has been used in all kinds of areas, and thus faces all kinds of users. However, there are two cooresponding problems.

2.1 Collaborative Filtering and Neural Network

The first one is whether the data is analyzable. The premise of that the recommender system could work properly largely depends on the accuracy of the input of the user's preference. However, the explicit feedback that is easy to analyze, for instance, the ratings, are actually only a small fracture of the data. There are also a large amount of implicit feedback, for instance, the purchasing history and browsing history. The features of the implicit feedback include the absence of negative feedback, the large noise and the demand of appropriate measures [1]. To analyze the implicit feedback, the collaborative filtering method is used. [1] designed a model which transforms the raw observations into two data, one is p measures how does user like the product, and the other one c measures how reliable p_i is. Through using this collaborative filtering, the implicit feedback becomes more analyzable and thus could give more accurate predictions. In addition to implicit feedback, the scale of the data and the freshness of the product could also bring challenge to the recommender system, for instance, the Youtube video recommendation [2]. Since the amount of videos in the Youtube is enormous, and that there are new videos everyday, and the interest of the user is also constantly changing, therefore the authors apply deep learning methods to recommendation systems by embedding implicit feedback together with additional information like user's demographic status and reframing recommendation as an extreme classification problem.

These previous works have shown the significance of the implicit feedback, for it contains large amount of information, and successfully analyzing the implicit feedback could largely increase the accuracy of the recommender system. Multiple methods are used to deal with implicit feedback. [3] uses deep neural networks to generalize the unseen feature combinations through "low dimensional dense embeddings learned for the sparse features" [3]. Another popular method is matrix factorization (MF)[4]. The main concept of the matrix factorization is that it models the user feature and the item feature into the same latent space so that they are under the same measure, thus the interaction of the user and the item could be evaluated using the inner product of the feature vector of the user and the item. Building on these foundations, [5] proposed a more general framework named Neural Collaborative Filtering (NCF), which replaces the inner product of the MF method to a neural network architecture. The NCF model is more generic, and by using multi-layer perceptron, the NCF is endowed with non-linearities, and it proved that the MF is also a special form of the NCF [5]. In our paper, we will mainly incorporate the NCF model to our analysis.

2.2 Fairness

The second is whether the model is fair. In addition to providing rankings based on user preferences, recommendation systems also have the responsibility to ensure fairness to both the users and items on the platform. Inappropriate gaps of ranking performances among different subgroups of users should be avoided, as well as systematic under-representation of items with similar features. These requirements may seem necessary at first glance, but they are critical components of the overall performance of recommendation systems. In the music and film industry, for instance, the awareness of consumers on new content greatly impacts market performance where recommendation systems contribute by providing a fair exposure of such content. Compared with classification problems where fairness is better studied with popular approaches like FPR- and FNR-rate matching [6], in recommendation systems fairness is harder to evaluate and improve due to reasons multi-fold. Fairness involves multi-task learning where the relationship between goals may influence prediction accuracy and thus the objectives must be balanced [7]. The shifting tastes of users and the inflow of new items calls for temporal dynamic analysis, and that is challenged by the scale of recommendation platforms, scarce user-item interactions and possible correlations between different characteristics [8]. A good amount of existing literature either investigates fairness during the pointwise prediction phase that predicts how much a particular user would like a particular item, or investigates fairness of final rankings that are often unpersonalized and impractical. [9] adopts a pairwise metric which directly corresponds to final ranking performance. Together with pairwise regularization in the algorithm, fairness among subgroups is enhanced for both pairwise metrics and pointwise metrics in experiment. [10] introduces a probabilistic ranking framework for ranking construction that is governed under several constraints of group fairness and individual fairness. The constraints are to ensure appropriate exposure allocation and the framework enables a flexible usage of different kinds of fairness constraints.

3 Method

In our project, we propose a new algorithm for recommendation systems called DNCF. it applies the same architecture as NFCF but is improved in the debiasing method and regularization. The DNCF enables debiasing for not only binary subgroups but also multi subgroups, and it has been experimented on different levels of regularization.

3.1 Baseline model

In our project, we will primarily adapt the NCF model from from [5] and [11]. In addition, since NCF model is an improvement of the MF model and under specific parameter setting the NCF model is equivalent to the MF model, in the experiments we also include the MF model in our baseline.

3.1.1 NCF

In the tradional Matrix Factorization (MF) model, the user embedding vector and the item embedding factor are mapped into the same latent space, and use inner product the embedding vectors to interaction between the user and the item. However, [5] replaced the inner product with a deep neural network to access the interaction, and the model is called the Neural Collaborative Filtering (NCF). According to [5], the NCF model has achieved satisfactory improvement comparing the the ordinary MF model.

3.1.2 NFCF

Although the NCF model has a satisfying overall performance, the biased data could produce unfair performances among some subgroups, for instance, among different genders. Thus, [11] introduced a Neural Fair Collaborative Filtering (NFCF) model to ensure the fairness of the NCF model. In [11], the authors introduced two fairness metrics, ϵ_{mean} and U_{abs}, and used these two metrics to regulate the unfairness created by biased data. According to [11], some non-sensitive biased data, like female user tends to access Barbie Doll page more, will influence the user embedding, and thus influence the recommendation of the sensitive data, like the job recommendation page. Using the pre-trained NCF model, the NFCF model also debiased the embeddings, and then input the debiased user embedding and parameters into an independent model designed for sensitive items, and use the fairness penalty to fine tune the parameters.

3.1.3 Our proposal: DNCF (debiased NCF)

However, the [11] only discussed how to revise the NCF model to make it fair among gender subgroups, which is a binary case. [11] didn't generalize the discussion of the NFCF model to a multi-subgroup case. Our concern is whether it is possible to apply the NFCF to other subgroups, especially multi-subgroups, for instance, the different age groups. In this project, we focus on how to use NFCF model to deal with the unairness among multi-subgroups and also conduct experiment on another binary subgroup of user age. In addition, we also tested DNCF under different levels of regularization.

4 Experiment

4.1 Dataset

In this project, we use the 1M-dataset from MovieLens. This dataset is the same as the dataset used in [11], which "contains 1 million ratings of 3,900 movies by 6,040 users who joined MovieLens" [11]. Different than [11], in addition to gender division of users, we create new labels for user by dividing them into binary and multi subgroups of age. In this regard we expand from binary subgroup debiasing to multi subgroup debiasing. In the original dataset, the users have already been divided into 7 age groups. The age attribute of a user denotes the minimum age of the age group this user belongs to, which ranges from 1 to 56. We use this default division as our multi subgroup for age, but we drop the subgroup of 1-18 due to very limited size of data. For the binary subgroup of age, we combine the 18-25 and 25-35 division from the multi subgroup into one age group that we call young adults, as well as combining the rest of the multi subgroups to another age group that we call middle-aged. This method of splitting results in a desirable proportion for the binary subgroups as both subgroup account for about 50% of the total data. We experiment with DNCF on all 3 sets of subgroups.

4.2 Experimental Settings

4.2.1 Models

In this project, we will compare the overall performance and fairness of the MF model, NCF model and NFCF model. To compare their performances, we first run pre-training version of MF and NCF. In the pre-training process of recommending movies to users, we train the user embedding and item embedding separately for the MF and NCF model. Afterwards, we run embedding debiasing files to improve the user embedding according to subgroups. The debiasing process applies the fairness penalty to obtain a "bias direction" vector [11]. According to [11], the debiasing procedure only needs to be performed for the user embedding, for there is "no additional fairness penalty in the object function"[11]. When we run the second task of recommending careers to users, we stop the user embedding from updating by setting the back propagation of its weight to False. With the debiased user embedding being static throughout training of the second task, we are able to fit the sensitive item of careers to this improved user embedding. Note that with the static debiased user embedding, other types of items can also be fit to it easily. This is the main purpose of performing a pre-training task and obtaining a debiased embedding afterwards. In [11], the embedding vector for a subgroup, say M, is defined to be

$$v_M = \frac{1}{|M|} \sum_{m \in M} v_m$$

where $\{v_m\}_{m \in M}$ are the embeddings of the users in subgroup M. Defining the same for subgroup N, the bias vector of binary subgroup M and N is given by

$$v_B = \frac{v_N - v_M}{||v_N - v_M||}$$

However, in our case when studying the multi-subgroups, this definition is not available, since this definition would give a bias vector for every pair of subgroups, and it would be too confusing

for computation. Therefore, in our project, when performing debias on the multi-subgroups, we use the following definition of bias vector:
For n subgroups $S_1, S_2, ..., S_n$ with average embeddings $v_{s_1}, v_{s_2}, ..., v_{s_n}$, we first obtain the average of these embeddings:

$$v_{avg} = \frac{1}{n} \sum_{i=1}^{n} v_{s_i}$$

So the bias vector is given by

$$v_B = \frac{\sum_{i=1}^{n} (v_{s_i} - v_{avg})}{|| \sum_{i=1}^{n} (v_{s_i} - v_{avg}) ||}$$

Then the de-bias is the same as in [11]. For each user embedding p_u, we perform de-biasing according to:

$$p'_u = p_u - (p_u \cdot v_B) v_B$$

where the p'_u is the de-biased user embedding. In the last step, we perform fine tuning of model parameters in the neural network. The fairness metric ϵ_{mean} is added to use as a fairness penalty to de-bias the demographic bias in sensitive items. The is the basic version of regularization adopted by [11], which we expand on by allowing to use different versions of subgroups as the fairness penalty, and also by allowing for stronger regularization by using multiple penalty terms from different subgroup measures.

4.3 Parameter Setting

For the deep neural networks we use a embedding size of 128. For the NCF model since it concatenates user embedding and item embedding in the first hidden layer, the actual embedding size is 256, while for MF the embedding size is 128. We set universal batch size of 256, learning rate of 0.001. The hidden layers are [$embedding_size$,64,32,16] for both models, and in NCF we apply Relu activation for hidden layers and Sigmoid activation for output layer. During the training of the first task, we set number of epochs to be 25 for both models. Since for the second task we load the pre-trained state dictionary for NCF, the epoch is 10 for NCF and 25 for MF. When updating the weights of neurons, we use Adam optimizer (adaptive gradient descent optimization) with lr=0.001 and slight weight decay of 1e-6. For every positive sample in the training data, we randomly select 5 negative samples along with it to avoid an excess of negative samples which will greatly degrade model performance and add to running time.

4.4 Fairness Metric

In this project, we adapt the two fairness metrics from [11], average epsilon and absolute unfairness. The specific definitions are given below:

1. The first metric is the average epsilon. In order to explain average epsilon, we first need to introduce the differential fairness and define the following variables:

   Let $M(x)$ denote the recommender system, which means $M(x)$ takes an input x and return an outcome y. The differential fairness is used to "ensure equitable treatment for all protected groups" [11], and $M(x)$ is said to be "$\epsilon - differentiallyfair(DF)$ with respect to (A, Θ) if for all $\theta \in \Theta$ with $x \sim \theta$ and $y \in \text{Range}(M)$ if

   $$e^{-\epsilon} \leq \frac{P_{M,\Theta}(M(x) = y|s_i, \theta)}{P_{M,\Theta}(M(x) = y|s_j, \theta)} \leq e^{\epsilon}$$

   for all $(s_i, s_j) \in A \times A$ where $P(s_i|\theta) > 0, P(s_j|\theta) > 0$ "[11]. The smaller the ϵ is, more fair the model is. For each item, the $M(x)$ will have a corresponding value ϵ_i, and what the eventual metric to evaluate the fairness of the model is

   $$\epsilon_{mean} = \frac{1}{n}\sum_{i=1}^{n} \epsilon_i$$

   Note that here the ϵ are the DF measure for sensitive items, since we are using the sensitive items to adjust the fairness.

2. The second fairness metric is the absolute unfairness. The absolute unfairness U_{abs} measures the difference of the behaviours for the advantaged and disadvantaged users [11]. The formula for U_{abs} is given by

   $$U_{abs} = \frac{1}{N}\sum_{j=1}^{N} ||E_D(\hat{y}_{ui})_j - E_D(r)_j| - |E_A(\hat{y}_{ui})_j - E_A(r)_j||$$

   where $E_D(\hat{y}_{ui})_j$ and $E_A(\hat{y}_{ui})_j$ denote the average predicted score for the j-th item for disadvantaged users and advantaged users respectively, and $E_D(r)_j$ and $E_A(r)_j$ are the average score for the disadvantaged users and advantaged users respectively [11].

4.5 Performance Evaluation

For the performance evaluation, we evaluate the performance of the ranked list. We adapt the method from [11], as it takes too much time to rank all the list, we gradually increase the size of the ranked list until it matches the top_K parameter and perform evaluation on every iteration. To evaluate the performance, we use the average Hit Ratio and Normalized Discounted Cumulative Gain (NDCG) as the metric for the performance [11]. As [11] mentioned, "HR measures whether the test item is present in the top-K list, while the NDCG accounts for the position of the hit by assigning higher scores to hits at top ranks. We calculated both metrics for each test user-item pair and reported the average score". While HR and NDCG measures the performance for the accuracy of the recommendation, we use the average epsilon and absolute unfairness to evaluate the fairness of the model.

4.6 Regularization

In addition to debiasing the user embedding, we impose regularization during model training to improve the fairness of final ranked lists and to better fit the item embedding to the debiased user embedding. Apart from the usual loss term applied in gradient descent, we add in a second loss term of fairness penalty that reflects the average epsilon and the epsilon base. These two are related to the ϵ_{mean} in our definition of fairness. Building on the regularization of [11], we tested different penalty terms that are computed by evaluating the fairness among different kinds of subgroups of gender and age. Another extension is that we experimented with penalizing on multiple fairness terms. Firstly, we applied this measure expecting the regularization to simultaneously reduce bias and unfairness among different definition of subgroups. Secondly, this allows us to test the influence of the degree of regularization on the trade-off between the overall performance and the fairness. Notice that for different subgroups there is intrinsic difference in the level of their average epsilon, that multi-subgroup will naturally have higher unfairness compared with binary-subgroups. It is up to designers to choose universal weight or customized weights for the loss term. In this project we use an universal weight of 0.1 for all experiments.

5 Results

In this section, we present the evaluated performance and fairness for MF, NCF and various versions of our proposed DNCF model. The gender debiased and gender regularized DNCF is equivalent to the NFCF model in [11]. We also visualizes the user embedding and the exact item that are recommended to users. Our multi subgroup debiasing method is proved useful in improving fairness and is not imposing excessive impact on embedding that would cause over-correction. The DNCF model has displayed consistency in different settings and multiple combinations of debiasing and regularization are tested on it. We found generally speaking, stronger regularization better improves fairness in all subgroup divisions, and limits the performance more. This is consistent with previous literature, and in our case the increased regularization considerably improves fairness and only mildly impacts performance. When the target subgroup matches the debiasing subgroup, the model produces satisfactory results.

5.1 Performance and Fairness

	Pre-training Task			
Models	HR@10	HR@25	NDCG@10	NDCG@25
NCF	0.518	0.282	0.805	0.354
MF	0.546	0.297	0.837	0.371

Table 1: Performance of NCF and MF in pre-training task

Table 1 displays the performance of NCF model and MF model in the pre-training task of recommending movies. The performance of MF and NCF is similar in both HR and NDCG, with MF slightly outperforming NCF. This result is consistent with the one in [11] and show that our pre-trained NCF model is reliable for further debiasing and tuning.

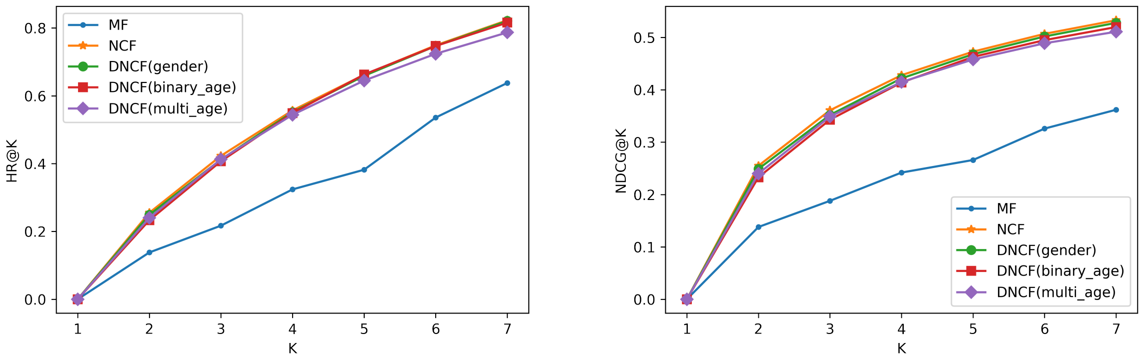

Figure 1: Model performance comparison

Figure 1 shows the performance of the MF, NCF, and three versions of our model DNCF in the second task of recommending a total of 16 careers. The DNCF models here use the same subgroup for debiasing and regularization, and all three subgroup of gender, binary age group and multi age group have been used. K stands for the top K items in the final recommendation list that we evaluate on. As K gets larger, we include more items and go further down the ranked list, so the performance metric gradually increases for all models. It is obvious that except MF which has apparently lower performance, NCF and all versions of DNCF have very similar performance. This shows that our debiasing and regularization method only impacts the model performance on a very small scale. Due to the penalty being naturally larger for multi subgroups, the DNCF(multi age) has a very small performance gap compared with other versions of DNCF.

Model	$\epsilon_{mean}(gender)$	$\text{U}_{abs}(gender)$	$\epsilon_{mean}(binary\_age)$	$\text{U}_{abs}(binary\_age)$	$\epsilon_{mean}(multi\_age)$	$\text{U}_{abs}(multi\_age)$
MF	0.119	0.049	0.028	0.014	1.843	0.322
NCF	0.107	0.011	0.151	0.030	0.877	0.047
DNCF	0.091	0.010	0.106	0.032	0.825	0.051

Table 2: Fairness for MF, NCF and DNCF

Table 2 evaluates the unfairness of MF, NCF and DNCF. The DNCF here is both debiased and regularized by gender, making it equivalent to the NFCF model in [11]. We can see that generally NCF and DNCF are more fair than MF. DNCF is better than NCF in terms of all three ϵ_{mean} metrics and is almost on the same level with NCF for U_{abs} metrics. This proves that DNCF is successful in reducing subgroup unfairness.

DNCF under different specifications							
Debias	Regularization	$\epsilon_{mean}(gender)$	$\text{U}_{abs}(gender)$	$\epsilon_{mean}(binary\_age)$	$\text{U}_{abs}(binary\_age)$	$\epsilon_{mean}(multi\_age)$	$\text{U}_{abs}(multi\_age)$
gender	gender	0.091	0.010	0.106	0.032	0.825	0.051
	binary_age	0.079	0.008	0.074	0.032	0.810	0.047
	multi_age	0.056	0.014	0.076	0.038	0.573	0.050
binary_age	gender	0.067	0.009	0.078	0.029	0.776	0.047
	binary_age	0.104	0.010	0.079	0.032	0.802	0.049
	multi_age	0.066	0.012	0.065	0.030	0.548	0.039
multi_age	gender	0.072	0.009	0.118	0.030	0.802	0.045
	binary_age	0.099	0.011	0.092	0.032	0.805	0.047
	multi_age	0.076	0.013	0.086	0.037	0.560	0.046

Table 3: Fairness of DNCF under different specifications of debias and tuning

For Table 3 we perform cross examination of 9 combinations of 3 debiasing settings and 3 regularization settings for our DNCF model. We compute the fairness metrics separately for all of the 3 subgroup measures. The results show that there is no apparent gap of unfairness among different versions of DNCF. This proves the consistency of our DNCF model that our debiasing method is not excessive. When one kind of subgroup is debiased, the other kinds of subgroup do not suffer from over-correction of user embedding. There is a general trend that for fairness metrics computed on a specific subgroup measure, the model that is debiased and regularized by that same subgroup measure would perform well. However, using different subgroup measures for debiasing and regularization is also viable as the resulting models also display improved fairness.

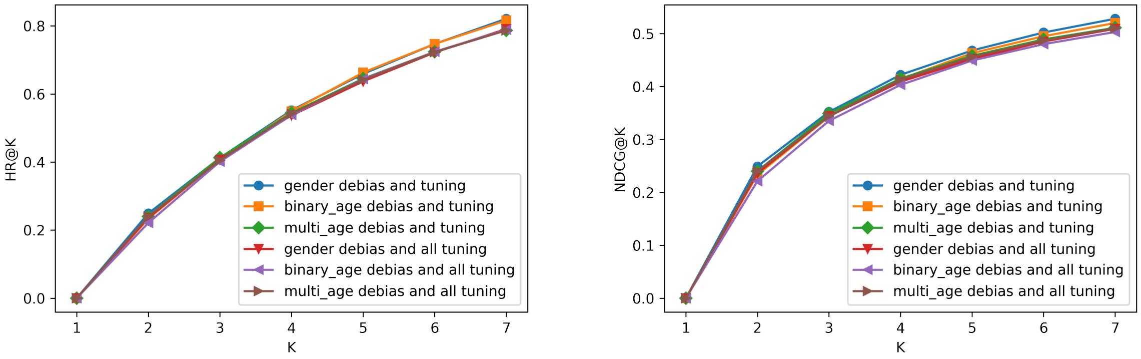

Figure 2: DNCF performance under different regularization

Figure 2 displays the performance comparison of DNCF models that use one penalty term for regularization and those that use all subgroup measures as penalty terms for regularization. We can see that again the results are very similar, with the stronger regularization versions only lacking behind a small amount. This shows that the added regularization terms have only mild impact on model performance, at least in our setting.

DNCF with regularization on all subgroup measures

Debias	$\epsilon_{mean}(gender)$	$\mathrm{U}_{abs}(gender)$	$\epsilon_{mean}(binary\_age)$	$\mathrm{U}_{abs}(binary\_age)$	$\epsilon_{mean}(multi\_age)$	$\mathrm{U}_{abs}(multi\_age)$
gender	0.046	0.014	0.064	0.036	0.543	0.046
binary_age	0.062	0.013	0.066	0.028	0.555	0.035
multi_age	0.053	0.011	0.072	0.036	0.550	0.043

Table 4: Fairness of DNCF under regularization of all subgroup measures

Table 4 evaluates the fairness of three versions of DNCF that are all regularized by three penalty terms from all subgroup measures. These models only differ in the subgroup used for debiasing. Compared with Table 3, we can see that the unfairness has considerably decreased for most of the metrics. Combining this result with Figure 2, it can concluded that added regularization is instrumental in improving subgroup fairness, while it only mildly impacts the model performance, at least in our setting. This demonstrates the value of our expansion over [11] that the regularization could be stronger for better overall recommendation.

5.2 Visualization

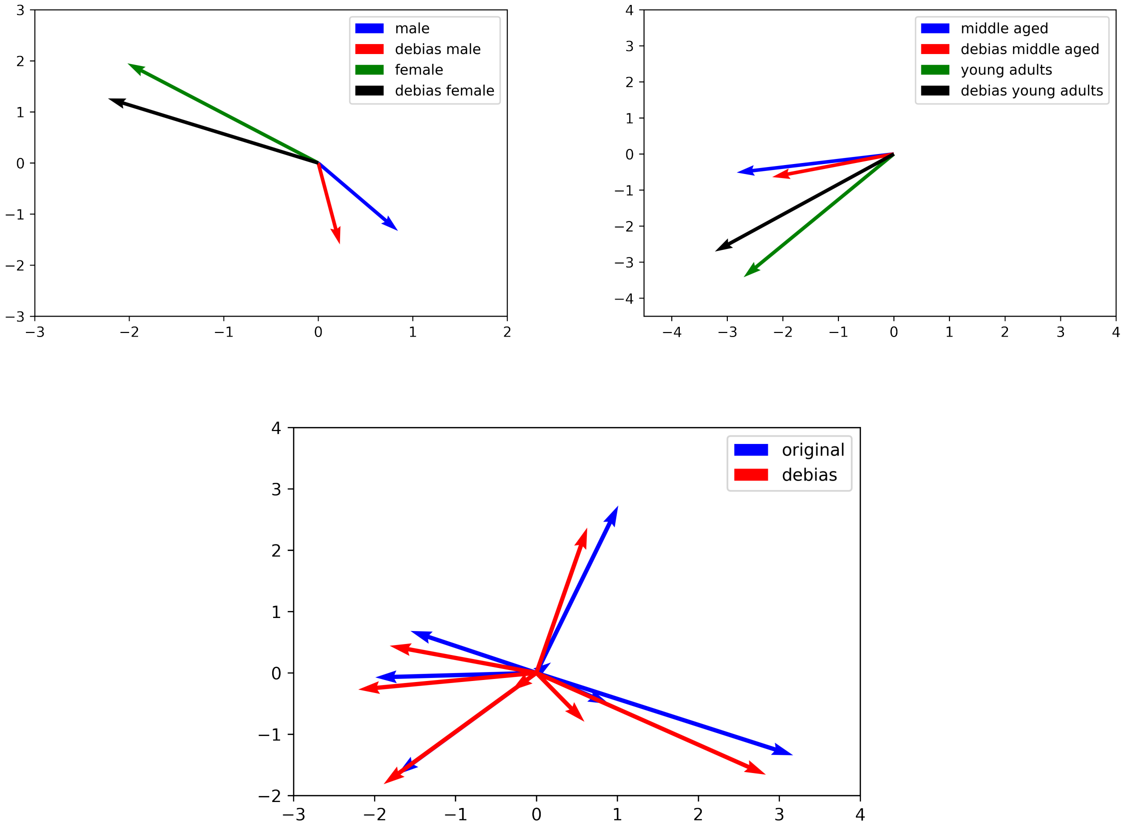

Figure 3: Visualization of embedding vectors

Figure 3 displays the effect of debiasing on the user embedding segmented by subgroups. The original embedding has a hidden dimension of 128 and we use the Principal component analysis (PCA) to reduce the dimension and visualize it on 2-D plots. For binary subgroups in the first two graphs, the original embedding is in blue or green, while the debiased embedding is in red or black. From the first two graphs we can see that our debiasing method has reduced the difference of directions in the embedding vectors, that the debiased vectors are closer to each other than the original vectors. This implies that the subgroup embedding have become more similar to each other and the debiasing is successful. For multi subgroup debiasing shown in the third graph, due to the number of vectors it is hard to make embedding more similar in terms of vector direction, but our method has helped reduced the variance of vector length. As shown in the section above this could also improve the fairness of final ranked list.

Table 5 and Table 6 visualizes the exact career being recommended to different subgroups of users in terms of gender and binary age group. These careers are the top 5 most frequent item

DNCF(gender debias)		DNCF(binary_age debias)	
Male	Female	Male	Female
college/grad student	college/grad student	college/grad student	college/grad student
executive/managerial	executive/managerial	executive/managerial	executive/managerial
academic/educator	academic/educator	academic/educator	technician/engineer
technician/engineer	technician/engineer	technician/engineer	academic/educator
programmer	programmer	programmer	programmer

Table 5: Career recommendation for different genders

DNCF(binary_age debias)		DNCF(gender debias)	
Young Adults	Middle-aged	Young Adults	Middle-aged
college/grad student	college/grad student	college/grad student	executive/managerial
executive/managerial	executive/managerial	executive/managerial	college/grad student
technician/engineer	academic/educator	technician/engineer	academic/educator
academic/educator	technician/engineer	academic/educator	technician/engineer
programmer	programmer	programmer	programmer

Table 6: Career recommendation for binary age groups

among the first item of the recommendation list to these users. Here the DNCF models use the same subgroup for regularization as specified for debiasing. We can see that indeed when debiasing and regularization match our target subgroup, the fairness of DNCF is the best. In addition, the consistency of DNCF is again displayed as when the target subgroups deviates from our debiasing method, the users still get relatively fair recommendations.

DNCF(multi_age debias)					
18	25	35	45	50	56
college/grad student	executive/managerial	executive/managerial	executive/managerial	executive/managerial	executive/managerial
executive/managerial	college/grad student	college/grad student	college/grad student	academic/educator	college/grad student
technician/engineer	academic/educator	academic/educator	academic/educator	college/grad student	academic/educator
academic/educator	technician/engineer	technician/engineer	technician/engineer	technician/engineer	technician/engineer
sales/marketing	programmer	programmer	programmer	programmer	programmer

DNCF(gender debias)					
18	25	35	45	50	56
college/grad student	college/grad student	college/grad student	college/grad student	college/grad student	college/grad student
executive/managerial	executive/managerial	executive/managerial	executive/managerial	executive/managerial	executive/managerial
academic/educator	technician/engineer	academic/educator	academic/educator	academic/educator	academic/educator
technician/engineer	academic/educator	technician/engineer	technician/engineer	technician/engineer	technician/engineer
programmer	programmer	programmer	programmer	/	/

Table 7: Career recommendation for multi age groups

Table 7 visualizes the careers recommended by two versions of DNCF to users of different multi age groups. Same with above the debaising and regularization uses the same subgroup, and the careers are the top 5 most frequent item among the first item of the recommendation list to these users. Our proposed multi-subgroup debiasing is successful in improving fairness among the 6 user subgroups as the recommendations are very similar. We compare this to another DNCF trained with gender debiasing, and it is obvious that the binary subgroup measures do not work well in multi subgroup settings. As we can see for certain multi subgroups the diversity of recommendation is rather limited like for 50 and 56 age groups.

6 Discussion

One main challenge in this project is the replication of results in literature took. While the mechanism of NCF seems easy, the code implementation is actually highly complicated. Besides, expanding the scope of debiasing is also challenging. The original NFCF paper [11] only worked with binary subgroups, but we expanded the discussion to multi-subgroups, which is a generalized version that is applicable in all cases. In addition, since the original MF and NCF are implemented in the different environments using different data, putting them in the same context and incorporating them with our fairness metrics also imposed great difficulty. Our work has achieved satisfactory overall performance and fairness under the designed fairness metrics. Compared to [11], our work expanded the discussion to multi-subgroups, and in this project, subgroups for different ages. Our model, after adding the fairness penalty on the age subgroups, has the similar overall performance compared with NCF model, and outperforms the MF model, and the fairness is satisfactory compared to the original NCF and MF models. In addition, we also studied how the degree of regularization would influence the performances and fairness. However, there are also limitations for our project. The first is that we are using static user embedding. The changes to the user embeddings due to the registration or close of an account could influence the user embeddings, and thus influence the result. Therefore ideally, it should be a "dynamic embedding". But for the complexity of this scenario, we didn't discuss it in this project. Besides, the same as in [11], in each test we only fix one attribute, gender or age, but we didn't discuss the case when these two attributes overlap, that is, when the users are divided into male young, female young, male old and female old. The overlapping of the attributes could complicate the problem. For future studies, we will work on how to adapt the model to dynamic embeddings, and study the case when the attributes overlap.

7 Conclusion

In this project, we successfully improved fairness in recommender systems among different subgroups of users. We extended upon previous literature and proposed our own method of multi subgroup debiasing together with the DCNF model that incorporates this method. Our proposed method is proved instrumental in improving fairness in multi subgroups by our visualization of recommended careers. Moreover, it does not over-correct the embedding and avoids exerting apparent impact on the fairness of other subgroup measures. The DNCF model has displayed consistency in multiple specifications and the overall performance and fairness exceeds our baseline models. When the target subgroup matches the debiasing subgroup, the model produces satisfactory results. Our results agree with the literature that regularization serves as a trade-off between performance and fairness. The stronger regularization, the better fairness and worse model performance. In the setting of this study, the increased regularization considerably improves fairness and only mildly impacts performance.

References

[1] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *2008 Eighth IEEE International Conference on Data Mining*, 2008, pp. 263–272.

[2] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for youtube recommendations." New York, NY, USA: Association for Computing Machinery, 2016. [Online]. Available: https://doi.org/10.1145/2959100.2959190

[3] H.-T. Cheng, L. Koc, J. Harmsen, T. Shaked, T. Chandra, H. Aradhye, G. Anderson, G. Corrado, W. Chai, M. Ispir, R. Anil, Z. Haque, L. Hong, V. Jain, X. Liu, and H. Shah, "Wide deep learning for recommender systems," 2016.

[4] X. He, H. Zhang, M.-Y. Kan, and T.-S. Chua, "Fast matrix factorization for online recommendation with implicit feedback," 2017.

[5] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," 2017.

[6] Y. Bechavod and K. Ligett, "Penalizing unfairness in binary classification," 06 2017.

[7] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi, "Modeling task relationships in multi-task learning with multi-gate mixture-of-experts." New York, NY, USA: Association for Computing Machinery, 2018. [Online]. Available: https://doi.org/10.1145/3219819.3220007

[8] Y. Koren, "Collaborative filtering with temporal dynamics." New York, NY, USA: Association for Computing Machinery, 2009.

[9] A. Beutel, J. Chen, T. Doshi, H. Qian, L. Wei, Y. Wu, L. Heldt, Z. Zhao, L. Hong, E. H. Chi, and C. Goodrow, "Fairness in recommendation ranking through pairwise comparisons," 2019.

[10] A. Singh and T. Joachims, "Fairness of exposure in rankings," *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery Data Mining*, Jul 2018. [Online]. Available: http://dx.doi.org/10.1145/3219819.3220088

[11] R. Islam, K. Keya, Z. Zeng, S. Pan, and J. Foulds, "Debiasing career recommendations with neural fair collaborative filtering," 2021.