JOHANNES KEPLER
UNIVERSITY LINZ

Author / Eingereicht von
Christian Ganhör
K11911652

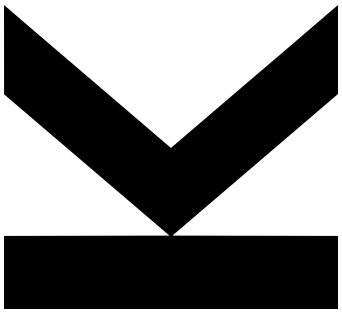
Submission / Angefertigt am
Institute for Computational Perception

Thesis Supervisor / First Supervisor / BeurteilerIn / ErstbeurteilerIn / ErstbetreuerIn
Univ.-Prof. Mag. Dipl.-Ing. Dr. **Markus Schedl**

Second Supervisor / ZweitbeurteilerIn / ZweitbetreuerIn
Dr. **Navid Rekab-saz**

October 2022

Unlearning Protected User Attributes in Recommendations with Adversarial Training



Bachelor Thesis

to obtain the academic degree of

Bachelor of Science

in the Bachelor's Program

Artificial Intelligence

JOHANNES KEPLER UNIVERSITY LINZ
Altenbergerstraße 69
4040 Linz, Austria
www.jku.at
DVR 0093696

JOHANNES KEPLER
UNIVERSITY LINZ

Preface

This thesis is based on a previous research paper [1], which was accepted at *ACM SIGIR 2022*. While some aspects are very similar (introduction, related work and model architecture), this work adresses some of the previous limitations and provides additional clarity and insight into adversarial training.

JOHANNES KEPLER
UNIVERSITY LINZ

Abstract

Collaborative filtering algorithms capture underlying consumption patterns, including the ones specific to particular demographics or protected information of users, e. g., gender, race, and location.

These encoded biases can influence the decision of a recommendation system (RS) towards further separation of the contents provided to various demographic subgroups, and raise privacy concerns regarding the disclosure of users' protected attributes.

In this work, we investigate the possibility and challenges of removing specific protected information of users from the learned interaction representations of a RS algorithm, while maintaining its effectiveness.

Specifically, we incorporate adversarial training into the state-of-the-art MULTVAE architecture, resulting in a novel model, *Adversarial Variational Auto-Encoder with Multinomial Likelihood* (ADV-MULTVAE), which aims at removing the implicit information of protected attributes while preserving recommendation performance. We conduct experiments on the MovieLens-1M and LFM-2b-DemoBias datasets, and evaluate the effectiveness of the bias mitigation method based on the inability of external attackers in revealing the users' gender information from the model. Comparing with baseline MULTVAE, the results show that ADV-MULTVAE, with marginal deterioration in performance (w. r. t. nDCG and recall), largely mitigates inherent biases in the model on both datasets. The introduction of two new metrics to evaluate user and group based biases introduced by recommender systems yield further insights into the effects of recommender systems and adversarial training.

JOHANNES KEPLER
UNIVERSITY LINZ

Kurzfassung

Algorithmen zur kollaborativen Filterung erfassen die zugrundeliegenden Konsummuster, einschließlich derjenigen, die sich auf bestimmte demografische Merkmale oder geschützte Informationen der Nutzer beziehen, z. B. Geschlecht, Rasse und Herkunft.

Diese kodierten Verzerrungen können die Entscheidung eines Empfehlungssystems (RS) dahingehend beeinflussen, dass die Inhalte, denen verschiedenen demografischen Untergruppen zur Verfügung gestellt werden, weiter voneinander getrennt werden, und sie können Bedenken hinsichtlich der Offenlegung geschützter Attribute der Nutzer aufwerfen.

In dieser Arbeit untersuchen wir die Möglichkeit und die Herausforderungen, spezifische geschützte Informationen von Nutzern aus den gelernten Interaktionsrepräsentationen eines RS-Algorithmus zu entfernen und gleichzeitig dessen Effektivität zu erhalten.

Konkret integrieren wir adversariales Training in die hochmoderne MULTVAE-Architektur, was zu einem neuartigen Modell, *Adversarial Variational Auto-Encoder with Multinomial Likelihood* (ADV-MULTVAE), führt, welches darauf abzielt, die impliziten Informationen geschützter Attribute zu entfernen und gleichzeitig die Qualität der Empfehlungen zu erhalten. Wir führen Experimente mit den Datensätzen MovieLens-1M und LFM-2b-DemoBias durch und bewerten die Effektivität der Methode zur Verringerung von Bias anhand der Tatsache, dass externe Angreifer nicht in der Lage sind, die Geschlechtsinformationen der Nutzer aus dem Modell herauszulesen. Im Vergleich zur Basislösung MULTVAE zeigen die Ergebnisse, dass ADV-MULTVAE mit einer geringfügigen Verschlechterung der Qualität (nach nDCG und Sensitivität) der inhärenten Bias des Modells in beiden Datensätzen weitgehend abschwächt. Die Einführung von zwei neuen Metriken zur Bewertung von nutzer- und gruppenbasierten Bias, die durch Empfehlungssysteme eingeführt werden, liefert weitere Erkenntnisse über die Auswirkungen von Empfehlungssystemen und adversarialem Training.

JOHANNES KEPLER
UNIVERSITY LINZ

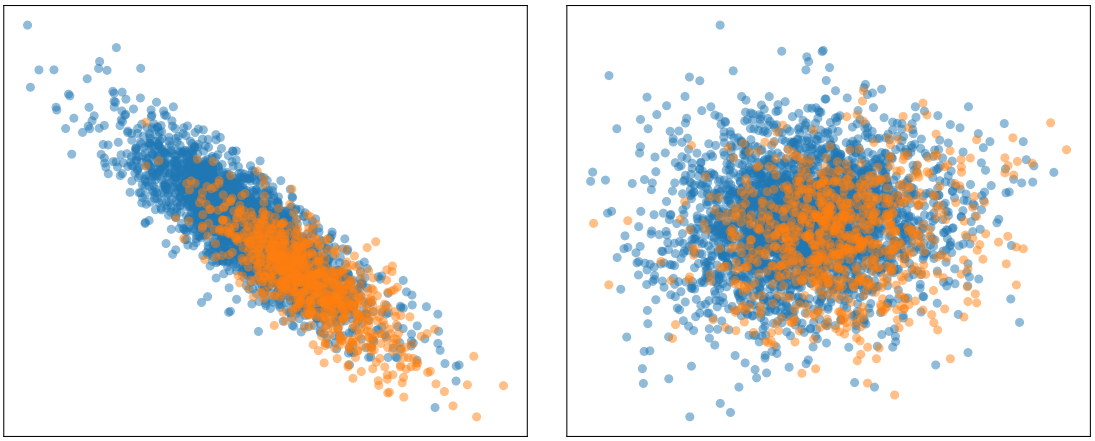
Contents

JOHANNES KEPLER
UNIVERSITY LINZ

List of Figures

List of Tables

1. Introduction



(a) MULTVAE **(b)** ADV-MULTVAE

Figure 1.1.: Output of an attacker network aiming to infer users' genders from the latent embeddings of the MULTVAE and ADV-MULTVAE models trained on LFM-2b-DemoBias [2] dataset. The blue and orange markers correspond to male and female users, respectively.

In recommender systems (RSs), collaborative filtering algorithms provide recommendations for users (consumers), primarily based on the collected user-item interactions, e. g., through listening to music tracks or watching movies. Among these algorithms, MULTVAE [3] learns to recommend items through decoding the variational encoding of user interaction vectors and has shown top results among a variety of deep neural network approaches [4]. While the interaction data does not explicitly contain information about protected user attributes such as gender, race, or age, a model may still encode sensitive information in its latent embeddings. This is depicted in Figure 1.1a, as the points regarding male and female users in a trained MULTVAE model form fairly separated clusters of users according to their genders.

These encoded biases in models can lead to strengthening "filter bubbles" based on the demographics of users [5, 6, 7, 8], and to intensifying the existing societal biases in data, thereby increasing unfairness of the RS [9, 2, 10, 11]. They can also raise privacy concerns regarding the disclosure of sensitive information from the recommendations or model parameters [12, 13, 14].

We approach this issue by proposing ADV-MULTVAE, a novel bias-aware recommendation model which enhances MULTVAE with adversarial training to reduce encoded biases. The ADV-MULTVAE model, while learning to provide effective recommendations, simultaneously forces its latent embeddings to become invariant with respect to a given protected attribute of the consumers. This results in reducing the distinguishability of the sub-populations in the model (as shown in Figure 1.1b), hence making the recommendation "blind" to the protected attribute while maintaining the model's recommendation performance. We particularly adopt MULTVAE, as it achieved top results among a variety of different deep neural network based approaches [4].

To assess the merits of our approach regarding both bias mitigation and recommendation performance, we conduct a set of experiments on the MovieLens-1m [15] and LFM-2b-DemoBias [2, 16] datasets covering the domains of movies and music, respectively. We focus on gender as the protected attribute and evaluate the accuracy and balanced accuracy of an attacker network to quantify the effect of bias mitigation. Furthermore, we introduce new user and group based metrics that give insight into how recommender systems introduce/enhance biases w. r. t. usual user tendencies. We assess the models' recommendation performance via nDCG, recall and coverage as utility metrics. ADV-MULTVAE successfully reduces inherent gender bias, while showing potential drawbacks on how it perceive user tendencies. A corresponding decrease in utility requires researchers and practicioners to trade-off between bias mitigation and model utility when employing and parameterizing ADV-MULTVAE.

The remaining work is structured as follows: We discuss related literature in Chapter 2. In Chapter 3 we introduce ADV-MULTVAE and our bias metrics, followed by Chapter 4, in which we present our experimental setup, and the datasets and metrics we use to evaluate our approach. Chapter 5 provides an analysis of our results. Finally, we conclude this work in Chapter 6.

2. Related work

2.1. Adversarial Training

As surveyed by Deldjoo, Noia, and Merra [17], adversarial training in combination with latent factor recommendation algorithms is investigated for various purposes by a few recent studies. In particular, Beigi et al. [12] propose a novel model based on Bayesian Personalized Ranking (BPR), which uses attacker networks to increase the model's privacy. In this model, the attacker networks aim to infer sensitive user information by looking at the output recommendations of the network, and the whole model is optimized such that no sensitive information can be inferred from the recommendations. Similarly, Zhang, Lemoine, and Mitchell [18] intend to mitigate biases of classifiers by utilizing adversarial networks, resulting in reducing the leakage of sensitive user attributes into the model predictions. In contrast to these studies, our proposed model aims to remove implicitly encoded sensitive information from its latent space rather than the output space. Moreover, unlike some approaches [19, 20] that apply filtering layers on top of their user embeddings to drop unwanted information, ADV-MULTVAE is trained with the objective that the information of the protected attributes is removed from the model in the first place.

Concerning bias mitigation in RSs, Zhu, Wang, and Caverlee [21] introduce the debiased personalized ranking model, in which the adversarial training aims to identify which item group, such as movie genre in the movie domain, the recommendation belongs to. This information is subsequently removed to mitigate *item popularity bias*. In contrast to this work, we study bias mitigation from the consumer side. More recently, based on adversarial training, Wu et al. [22] explore the mitigation of consumer bias in news recommendation, and several recent studies [10, 23, 24, 25] approach fairness in the representation of gender-related documents in information retrieval. Our work extends these

studies by introducing a novel bias-aware recommendation model based on variational autoencoders.

2.2. Bias metrics

Chen et al. [26] survey different types of biases as well as different approaches to mitigate them. Based on their characterization, our defined bias metrics are biases that can be observed in the RS results, i. e., the recommendations proposed to the users. Yao and Huang [27] introduce four metrics to measure difference in treatment between two separate user groups. These metrics are based on differences in predicted and actual ratings for specific user-item combinations. In comparison to their work, our bias metrics intend to shine light on how specifically recommender systems perceive users and how recommendations in general are adapted. Moreover, our metric are aggregated over individual users rather than individual items.

A separate work by Lesota et al. [28] focuses on analyzing the descripancies in popularity bias between different user groups (gende, age, ...). The authors define 7 metrics to measure and describe shifts in popularity distributions between recommended items and items users previously consumed, and evaluate different collaborative filtering algorithms based on them. In our work, we instead analyze whether previously expressed user group trends are maintained in recommendations. Moreover, we only focus on Variational Auto-Encoders (VAE) [29], with and without applied adversarial training.

3. Methodology

3.1. ADV-MULTVAE

In the following, we describe the architecture of our *Adversarial Variational Auto-Encoder with Multinomial Likelihood* (ADV-MULTVAE) model. We first provide an overview of the baseline MULTVAE, followed by explaining our adversarial extension. We finally describe the procedure of adversarial attacking used to assess the effectiveness of bias mitigation. Figure 3.1 depicts the outline of the proposed ADV-MULTVAE model.

MULTVAE The MULTVAE model consists of two parts: First, the encoder network $f(\cdot)$ receives the input vector $\boldsymbol{x}$ containing the interaction data of a user and infers a low-dimensional latent distribution. Considering a standard Gaussian distribution as the prior ($\mathcal{N}(0, I)$) and using the *reparameterization trick* [29], this distribution is characterized by $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ learnable vectors, from which the latent vector $\boldsymbol{z}$ is sampled. [1] The second part is the decoder network $g(\cdot)$, which aims to reconstruct the original input $\boldsymbol{x}$ from the latent vector $\boldsymbol{z}$ by predicting $\boldsymbol{x}'$. We refer to the loss function of MULTVAE as $\mathcal{L}^{\text{rec}}(\boldsymbol{x})$, defined below:

$$\mathcal{L}^{\text{rec}}(\boldsymbol{x}) = \mathcal{L}^{\text{MULT}}(g(\boldsymbol{z}), \boldsymbol{x}) - \beta\mathcal{L}^{\text{KL}}(\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\sigma}), \mathcal{N}(0, I)) \tag{3.1}$$

where $\mathcal{L}^{\text{MULT}}$ is the input reconstruction loss, and $\mathcal{L}^{\text{KL}}$ is the regularization loss aiming to keep the latent distribution of the encoder close to the prior, whose influence is adjusted by the hyperparameter β. We refer to Liang et al. [3] for more details.

[1] In short, reparametrization trick allows sampling the random variable $\boldsymbol{z}$ by reparameterizing the sampling process with an auxiliary stochastic variable, thereby maintaining the ability to perform back-propagation on $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$.

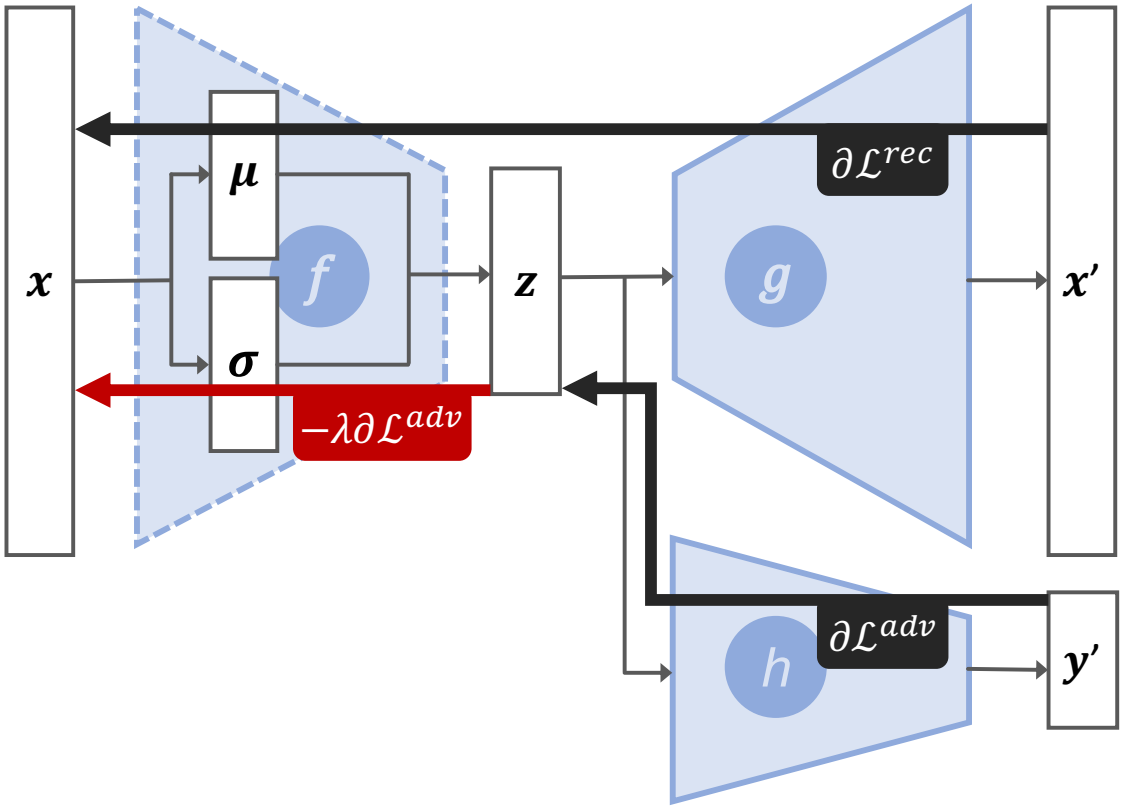


Figure 3.1.: Outline of ADV-MULTVAE. The bold arrows show the flow of gradients during backward pass, where the red color indicates the reversed gradient for learning latent embeddings (z) invariant to the protected attribute (y).

ADV-MULTVAE Our proposed model extends MULTVAE with an adversarial network, referred to as $h(\cdot)$. The adversarial network is added as an extra head over the latent vector and aims to predict from latent vector $\boldsymbol{z}$ a specific protected attribute of the user.

$h(\cdot)$ is typically a feedforward network, which is optimized with respect to the $\boldsymbol{y}$ vector containing the user's protected attribute as classification labels. The training process of ADV-MULTVAE aims to simultaneously remove the information of the protected attribute from $\boldsymbol{z}$, and maintain recommendation performance. To this end, the loss of the model is defined as the following min-max problem:

$$\arg\min_{f,g} \arg\max_{h} \mathcal{L}^{\text{rec}}(\boldsymbol{x}) - \mathcal{L}^{\text{adv}}(\boldsymbol{x}, \boldsymbol{y}) \tag{3.2}$$

where the loss of the adversarial network $\mathcal{L}^{\text{adv}}$ is defined as the cross-entropy loss ($\mathcal{L}^{\text{CE}}$) between the predicted and actual value of the protected attribute: $\mathcal{L}^{\text{adv}}(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{L}^{\text{CE}}(h(\boldsymbol{z}), \boldsymbol{y})$.

In fact, the loss defined in Eq. 3.2 aims to maximize the prediction ability of $h(\cdot)$ to discover all sensitive information when $\boldsymbol{z}$ is given, while it minimizes the encoded information in $\boldsymbol{z}$ concerning the protected attribute.

Considering the well-known complexities of optimizing min-max loss function [30], following previous work [10, 31, 32], we add a gradient reversal layer $grl(\cdot)$ [33] between $\boldsymbol{z}$ and the adversarial network $h(\cdot)$. During training, $grl(\cdot)$ acts as the identity function in the forward pass, while it scales the calculated gradient by $-\lambda$ in the backward pass. The $grl(\cdot)$ network does not have any effect on the model at inference time. We refer to the parameter λ as *gradient reversal scaling*. By employing $grl(\cdot)$ in the model, the overall loss in Eq. 3.2 can now be reformulated to a standard risk minimization setting:

$$\begin{aligned} &\arg\min_{f,g,h} \mathcal{L} = \mathcal{L}^{\text{rec}}(\boldsymbol{x}) + \mathcal{L}^{\text{adv}}(\boldsymbol{x}, \boldsymbol{y}), \\ &\mathcal{L}^{\text{adv}}(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{L}^{\text{CE}}(h(grl(\boldsymbol{z}), \boldsymbol{y}) \end{aligned} \tag{3.3}$$

This formulation enables optimizing the model through standard gradient-based loss minimization.

Adversarial Attacks. After training the model (whether MULTVAE or ADV-MULTVAE), we examine to which extent the information of the protected attribute remains in the

model, i. e., to which degree this information can still be recovered. To this end, once the training is complete, an attacker network $h^{\text{atk}}(\cdot)$ is introduced to the model, which aims to predict the protected attribute $\boldsymbol{y}$ from the latent vector $\boldsymbol{z}$. Similar to $h(\cdot)$, the attacker $h^{\text{atk}}(\cdot)$ is defined as a feedforward network. During training the attacker, all model parameters remain unchanged (are frozen) and only the attacker parameters are updated. The prediction performance of the attacker – relative to a random predictor – is used as a metric to quantify the degree of bias in the underlying model.

3.2. User Tendency Bias

In the following, we introduce user and group based metrics to evaluate recommender systems on the amount of bias towards items they introduce with respect to any binary user attribute.

Let $u \in \mathcal{U}$ be the set of users, $i \in \mathcal{I}$ the set of items in the dataset. Moreover, let some binary attribute a (e. g., $0/1$, $false/true$, $\dots$) be the attribute with respect to which the biases should be evaluated. We split the users of $\mathcal{U}$ based on the binary evaluation of attribute a, leading to the disjoint subsets $\mathcal{U}^{(a_f)}$ and $\mathcal{U}^{(a_t)}$, where a_f and a_t correspond to the negative ($false$) and positive ($true$) outcome of the evaluation, respectively. For simplicity, we say that users of subset $\mathcal{U}^{(a_f)}$ belong to group g_f, while users of subset $\mathcal{U}^{(a_t)}$ belong to group g_t.

Now, let r_{ui} indicate whether user u has interacted with item i.[2] For each item i we define its group preference, i. e., which group is more likely going to interact with it, as the difference of proportions of the users of both user groups that interact with the item:

$$\omega_i^{(a)} = \frac{\sum_{u \in \mathcal{U}^{(a_t)}} r_{ui}}{\#\mathcal{U}^{(a_t)}} - \frac{\sum_{u \in \mathcal{U}^{(a_f)}} r_{ui}}{\#\mathcal{U}^{(a_f)}}, \tag{3.4}$$

Using the proportion of users per group rather than absolute numbers leads to $\omega_i^{(a)} \in [-1, 1]$, allowing comparable results across datasets as well as interpretations on an item level. As an example, item group preference value close to -1 for some item $i \in \mathcal{I}$, i. e., $\omega_i^{(a)} \approx -1$, implies that users of group g_f are especially interested in the item, while

[2] Considering only whether a user has or has not interacted with an item allows broad application of the metric on a variety of data sources such as ratings and play-counts.

users of group g_t are not. On the other hand, a value close to 0 implies that users of both groups show comparable levels of interest.

To evaluate a recommender system's behavior, for each user $u \in \mathcal{U}$, we now aggregate the preference values of the set of items $\mathcal{I}_u^{(rec)} \subset \mathcal{I}$ recommended to them. For aggregation, we chose the median $med(\cdot)$ as it is a common and easily interpretable metric that is robust against outliers.

$$user\_tendency_u^{(a,rec)} = med\Big(\{\, \omega_i^{(a)} \mid i \in \mathcal{I}_u^{(rec)} \,\}\Big) \tag{3.5}$$

The result may be thought of as the item attribute tendencies a RS expects for some user w. r. t. attribute a to prefer, based on their history, and thus recommends specific items.

Similarly, one can calculate a user's actual (ground truth) item tendency by using $\mathcal{I}_u^{(hist)} \subset \mathcal{I}$, the set of items the user previously interacted with.

$$user\_tendency_u^{(a,hist)} = med\Big(\{\, \omega_i^{(a)} \mid i \in \mathcal{I}_u^{(hist)} \,\}\Big) \tag{3.6}$$

We want to highlight that, as different user groups may show different activity on e. g., streaming platforms [34], $user\_tendency_u^{(a,rec)} = 0$ and $user\_tendency_u^{(a,hist)} = 0$ do not necessarily imply neutral tendencies.[3] While there are normalization options available to achieve neutrality at $user\_tendency_u^{(a,rec)} = 0$, the added complexity might introduce unwanted biases. Thus, we settle for the less complex solution and compare the results with ground truth results instead.

Next, we introduce $tendency\_bias^{(a)}$, the average difference of the users' actual and RS perceived interaction tendencies.

$$tendency\_bias^{(a)} = \frac{1}{\#\mathcal{U}} \sum_{u \in \mathcal{U}} \Big(user\_tendency_u^{(a,rec)} - user\_tendency_u^{(a,hist)} \Big) \tag{3.7}$$

Values of relatively large magnitude indicate that a recommender system fails to recognize users actual tendencies, and thus introduces or enhances biases. On the other hand, values close to 0 imply overall unbiased recommendations, matching users actual tastes.

[3] We use the term *neutral tendencies* for describing tendencies of sets of items with an, on average, equal amount of both user group item attribute tendencies.

Similarly, $tendency\_bias^{(a)}$ can also be calculated for each of our disjoint user groups individually, enabling further analysis.

Finally, we define $tendency\_difference^{(a,rec)}$ by separately calculating $tendency\_bias^{(a)}$ for each of our disjoint user groups g_f and g_t, and taking their difference.

$$tendency\_difference^{(a,rec)} = \frac{\sum_{u \in \mathcal{U}^{(a_t)}} user\_tendency_u^{(a,rec)}}{\#\mathcal{U}^{(a_t)}} - \frac{\sum_{u \in \mathcal{U}^{(a_f)}} user\_tendency_u^{(a,rec)}}{\#\mathcal{U}^{(a_f)}} \quad (3.8)$$

This reflects how similarly a recommender system treats users of the different user groups. Again, conclusions can only be made by comparing the result to the ground truth, the similarly derived $tendency\_difference^{(a,hist)}$.

4. Experiments

In this chapter, we describe the setup of our experiments.

4.1. Datasets

We evaluate our approach on four standardized datasets containing user-item interactions as well as partial demographic information of their users:

- **ML-100K**[1] and **ML-100K**[2] **[15]** contain ratings of users on movies as well as the users' gender, age and occupation information. We binarize the interactions by setting the values of the rated items to one, and the rest to zero. Finally, we only keep the users that rated at least 5 movies, and the movies with at least 5 user interactions.
- **LFM-SMALL** and **LFM-BIG**, two subsets of the LFM-2B-DEMOBIAS[3] dataset [2], which provides a collection of music listening records of users, for whom partial demographic information (gender, age, country) is available. For our experiments, we only keep user-item interactions from the year 2019 with a play count of at least 2 and binarize the interactions. Moreover, for computational reasons, 10,000 and 100,000 tracks for LFM-SMALL and LFM-BIG, respectively, are randomly sampled from the data. Finally, we only keep users with all geographic information available and with at least 10 track interactions, and tracks that are listened to at least 10 times.

[1] https://grouplens.org/datasets/movielens/100k
[2] https://grouplens.org/datasets/movielens/1m
[3] http://www.cp.jku.at/datasets/LFM-2b

Dataset	Users		Items	Interactions	Avg. Interactions	Density
ML-100K	All	943	1,349	99,287	105.29	0.0780
	Male	670		73,824	110.19	0.0817
	Female	273		25,463	93.27	0.0691
ML-1M	All	6,034	3,125	574,376	95.19	0.0305
	Male	4,326		429,039	99.18	0.0317
	Female	1,708		145,337	85.09	0.0272
LFM-SMALL	All	5,130	5,677	165,003	32.16	0.0057
	Male	4,222		138,618	32.83	0.0058
	Female	908		26,385	29.06	0.0051
LFM-BIG	All	7,603	62,617	1,845,963	242.79	0.0039
	Male	6,165		1,541,815	250.09	0.0040
	Female	1,438		304,148	211.51	0.0034

Table 4.1.: Statistics of the datasets used in our experiments.

The statistics of the datasets are reported in Table 4.1. For all datasets, we focus on the users' gender as the protected attribute for our experiments.[4] We want to highlight that, while the datasets are only from two distinct data sources, using different subsets allows us to experiment on different dataset sizes (w. r. t. number of users, items and interactions) as well as on different dataset densities. As shown in the results section (section 5), the recommender systems perform differently on the individual datasets.

4.2. Data Splits

Following the setting of Melchiorre et al. [2], we apply a user split strategy [35]. In this setting, the users (and their corresponding interactions) are split into 5 folds for cross-validation, where 3 folds make up the training set, and 1 fold each makes up the validation and test set. Note that our training pipeline (see section 4.4 below) requires us to have an additional validation split next to the common train-test splits to ensure an accurate evaluation on unseen data. In case items do not have any interactions in the training

[4]The provided gender information of the users in the datasets are limited to female and male, neglecting the more nuanced definition of genders. Despite this limitation, the introduced model is generic and can be applied to non-binary settings too.

set, they are dropped from all splits in a run to not negatively influence the evaluation metrics. For the training set, we further perform random upsampling of female users (as the minority group) to achieve a balanced dataset, which supports bias mitigation in models [2]. For validation and testing, the interactions in each set are further split: 80% are used as model input, the remaining 20% for calculating the evaluation metrics.

4.3. Evaluation

4.3.1. Utility metrics

We use two popular user-based recommendation performance metrics: **recall@k**, namely the fraction of relevant items, in the top k recommended items, and **nDCG@k**, which weights the relevance of the top k recommended items based on their ranking positions. As common, we set the cut-off threshold k to 10. Additionally, we measure the effectiveness of the models in terms of system bias mitigation using the **accuracy (Acc)** and **balanced accuracy (BAcc)** of the adversaries and attacker when predicting the user's gender. We use BAcc as a proper metric in imbalanced classification settings [36]. It reports the average recall per class (female/male), where a value of 0.50 indicates a fully debiased network. As a general indicator of the diversity of recommendations, we further calculate **coverage**, the proportion of all unique items that are recommended to the users.

4.3.2. User Tendency Bias

We measure different user based biases as previously described in section 3.2, where we use *gender* as the binary attribute a, while acknowledging the previously mentioned shortcommings. Female and male users will be assigned the labels $false$ and $true$, respectively. As basis, to mitigate potential negative effects due to normalizations, the item attribute preferences $\omega_i^{(a)}$, $i \in \mathcal{I}$ (equation 3.4) are calculated on a balanced subset of all users in a dataset. Here, the overrepresented group w. r. t. gender, the group of male users for our datasets, is downsampled. We repeat user sampling and calculation of $\omega^{(a)}$ 5 times for a better estimation, calculate all following bias metrics for each sampling individually, and only aggregate them by taking the average on the final reports.

Similar to the utility metrics, for calculating the recommenders perceived user item tendency $p_u^{(a,rec)}$ (equation 3.5), the set of recommended items $\mathcal{I}^{(rec)}$ is of size 10. For the users actual tendency $p_u^{(a,hist)}$ (equation 3.6), we use the complete set of relevant items.

We report the RS's utility, the system bias mitigation and the user bias results as the average over all test sets' results across cross-validation folds.

4.4. Models and Training

We train the MULTVAE and ADV-MULTVAE models for 200 epochs on all datasets, and store them every 20 epochs to allow the analysis and visualization of the gradual changes in model behaviour. Due to the difficulty of jointly optimizing the subnetworks of ADV-MULTVAE (encoder+decoder network often start to overfit while adversaries are still learning), we employ PyTorch's [37] *ReduceLROnPlateau*[5] learning rate (LR) scheduler. This scheduler reduces the learning rate for the encoder+decoder network once their nDCG on the validation set hits a plateau, or already starts decreasing again and helps in partially reducing the negative effects of long training after the encoder+decoder network has reached its peak performance. A minimum enforced LR ensures that, while the adversarial network improves (and modifies the encoders weights), the model's utility does not decrease.

Although the selection of the best MULTVAE model configuration is mainly based on the validation nDCG results at the last training epoch, we also consider the training curves w. r. t. speed of convergence and stability. We perform a hyperparameter search over embedding size, parameter β, various dropouts, optimizer settings (LR and weight decay) and scheduler settings. For ADV-MULTVAE, we use the best MULTVAE configuation, and additionally perform a hyperparameter search over adversary size and number of adversaries, input dropout, adversary optimizer settings and gradient reversal scalings λ. The best ADV-MULTVAE model is considered to be the one with the best BAcc results, i. e., closest to 0.5 to imply inability of attribute prediction on the validation set at the last epochs of training. Concretely, we consider the average over the last 5 training epochs for more stability against fluctuations. All used hyperparameters as well as the finally

[5] `https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.ReduceLROnPlateau`

selected values (marked in bold) for MULTVAE and ADV-MULTVAE can be found in table A.1 and in table A.2, respectively.

5. Results

In the following, we provide an extensive overview on the results of MULTVAE and ADV-MULTVAE, and discuss the implications and potential trade-offs of applying adversarial training to recommender systems.

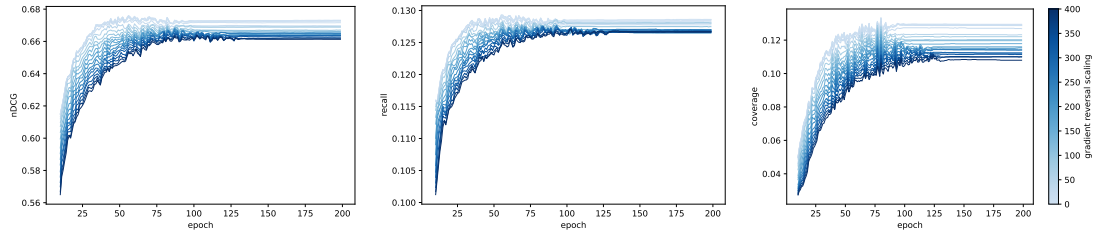
5.1. Recommender system utility

The right column of Figure 5.1 reports the influence of the gradient reversal scaling λ on the nDCG over the course of training. For all datasets, a similar observation can be made: λ and nDCG negatively correlate, i. e., the greater factor is, the stronger it negatively effects the utility of the recommender system. Note that $\lambda = 0$ corresponds to the MULTVAE model, to which we will also refer to as baseline model. As can be observed when looking at the difference in nDCG at the end of training, i. e., epoch=200, the nDCGs significantly drop in the range of $0.01 - 0.02$ for the most adversarially trained models (highest λ) in comparison with the baseline model.

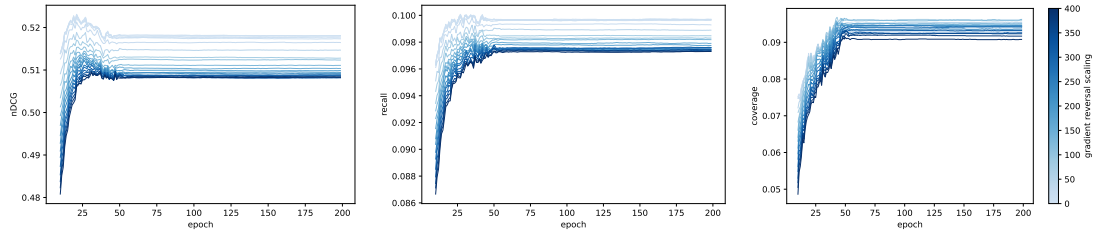
One can also observe that the nDCGs do not increase further after approximately 100 epochs, suggesting that the training models' recommender parts (encoder and decoder networks) are complete. As the nDCG may even start decreasing due to overfitting, e. g., see ML-1M (b), the LR scheduler reduces the learning rate used for updating the encoder and decoder networks to mitigate further deterioration.

Similarly, while not to the same extend, the recall (center column of Figure 5.1) is also significantly negatively impacted by large λ with recall decreases in the range of $0.001 - 0.005$.

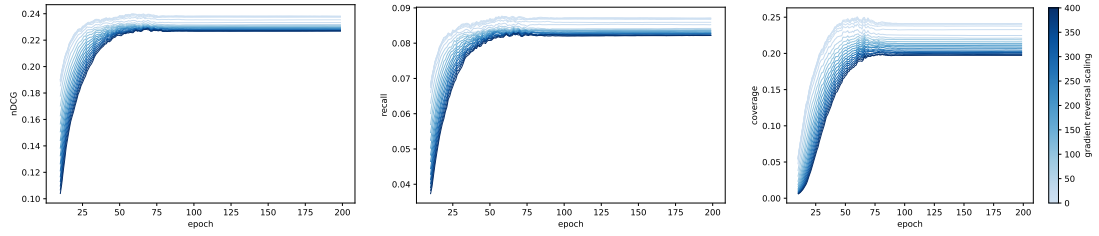
Moreover, this is also the case for the coverage of recommended items. Similarly to the nDCG, a strong negative correlation between coverage and gradient reversal scaling can



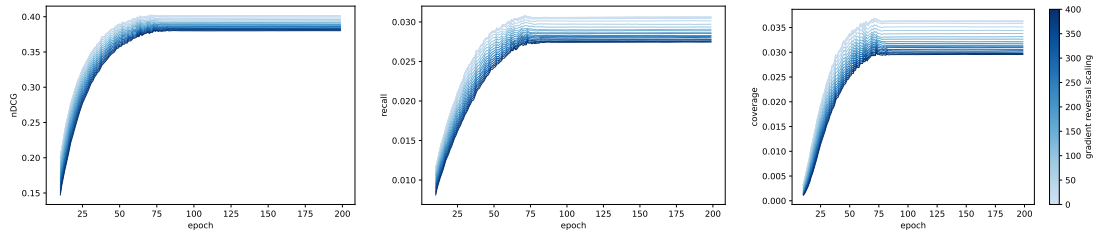
(a) ML-100K



(b) ML-1M



(c) LFM-SMALL



(d) LFM-BIG

Figure 5.1.: Influence of adversarial scaling on nDCG (left), recall (center) and coverage (right) over the course of training.

Dataset	Model	nDCG	recall	coverage
ML-100K	MULTVAE	0.674	0.129	0.129
	ADV-MULTVAE	0.664†	0.128	0.110
ML-1M	MULTVAE	0.518	0.100	0.093
	ADV-MULTVAE	0.509†	0.098†	0.092
LFM-SMALL	MULTVAE	0.239	0.087	0.243
	ADV-MULTVAE	0.230†	0.084†	0.204
LFM-BIG	MULTVAE	0.399	0.030	0.037
	ADV-MULTVAE	0.385†	0.029†	0.032

Table 5.1.: Comparison of utility metrics between baseline MULTVAE and maximally adversarially trained counterpart ADV-MULTVAE ($\lambda = 400$), both evaluated on the test set. The † indicates a significant decrease in nDCG and recall of ADV-MULTVAE in comparison to MULTVAE.[1]

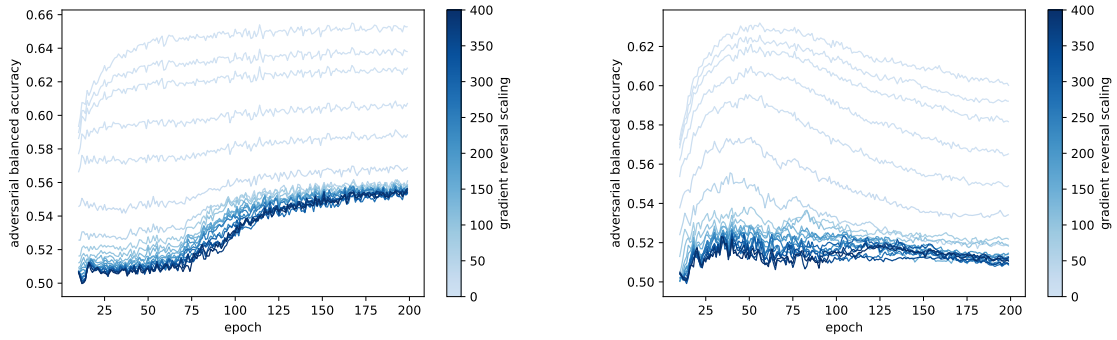
be observed on all datasets (drop in the range of $0.01 - 0.05$). The resulting recommendations of an adversarially trained RS are thus much less diverse than its baseline counterpart. While this prompts questions on the enhancement adversarial training has on other biases (such as popularity bias [28]), this will not be pursued further.

We provide an overview of all utility metrics discussed above in Table 5.1.

5.2. Adversarial and attacker accuracy

Figure 5.2b shows how well different adversaries are able to determine the users sensitive attribute (gender) on the LFM-SMALL dataset over the course of training, evaluated on the validation set. Initially, the classifiers perform nearly at random (BAcc ≈ 0.5). During training, especially for the baseline model MULTVAE ($\lambda = 0$), the accuracy gradually increases until it reaches its peak performance (BAcc ≈ 0.585), from where the accuracy decreases again. Comparing the accuracies to the ones on the training set (see Figure 5.2a), overfitting of the adversaries can be observed. Although this may look unintended, having stronger adversaries than required for low λ is actually necessary. Otherwise, for

[1] We omit testing for significance on coverage as it is calculated on the whole test set, which would require running the experiments many times to gather sufficient data.



(a) train set **(b)** validation set

Figure 5.2.: Adversary accuracy over the course of training for the LFM-SMALL dataset on the train (left) and validation (right) set.

increasing λ, the adversarial components may lack the power to recognize encoded sensitive user information when the encoders learn to hide them.[2]

Thus, although the performance of the adversaries may indicate that there is hardly any sensitive user information left in the models' latent spaces, only the results of a separately trained classifier networks (attackers) on the latent spaces can report the actual amount of encoded information [12]. Figure 5.3 reports the BAcc of attackers on ADV-MULTVAE models at different stages of training. While for the baseline MULTVAE, the attacker accuracy remains consistently high at around BAcc $= 0.64$, for all other models, i. e., $\lambda > 0$, the attacker accuracies continuously decrease toward BAcc $= 0.5$, indicating difficulty for the attackers in determining protected user attributes. This, in fact, highlights the need of long training runs for ADV-MULTVAE, even after the model's utility has reached peak performance. Otherwise, the mitigation of sensitive user information in the model may not be completely exhausted.

For selecting the most debiased model, we therefore take a closer look at the final training epoch, as visible in Figure 5.4. Here, while gradient reversal scaling and accuracy negatively correlate, the most debiased model may not necessarily be the one with the highest scaling factor. As an example, take a look at the curve of LFM-BIG, where the model with $\lambda = 200$ is the most debiased one. For in-production use, selecting such a model may be more advantageous than selecting ones with higher scaling values, as it

[2] An individual hyperparameter search for each gradient reversal scaling would also be an option, however this is not worth the additional computation costs.

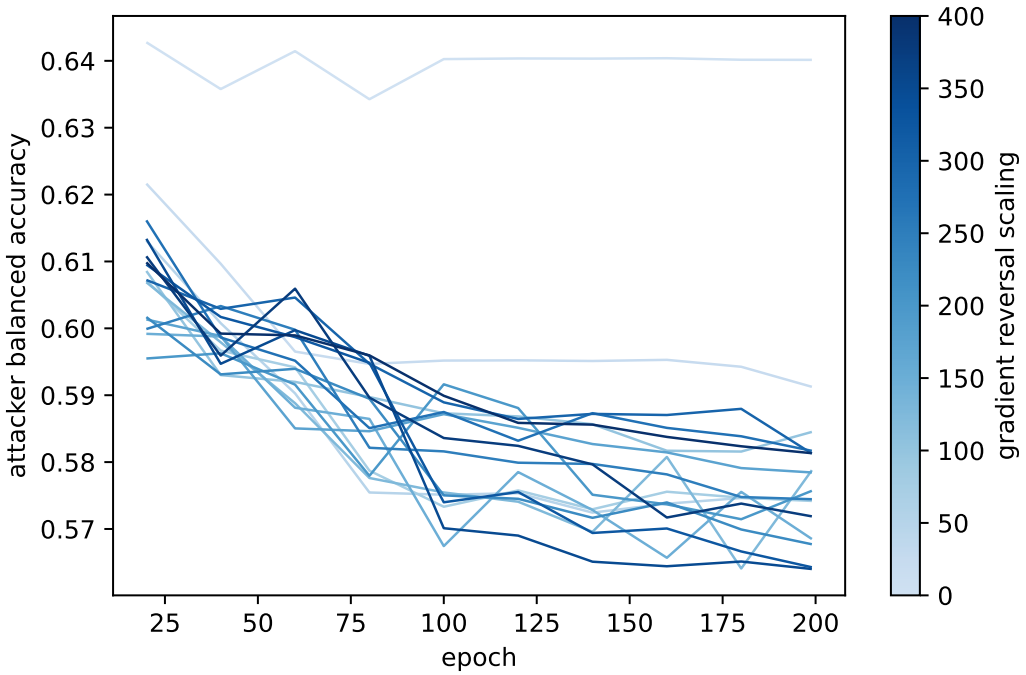


Figure 5.3.: Attacker accuracy over the course of training for the LFM-SMALL dataset.

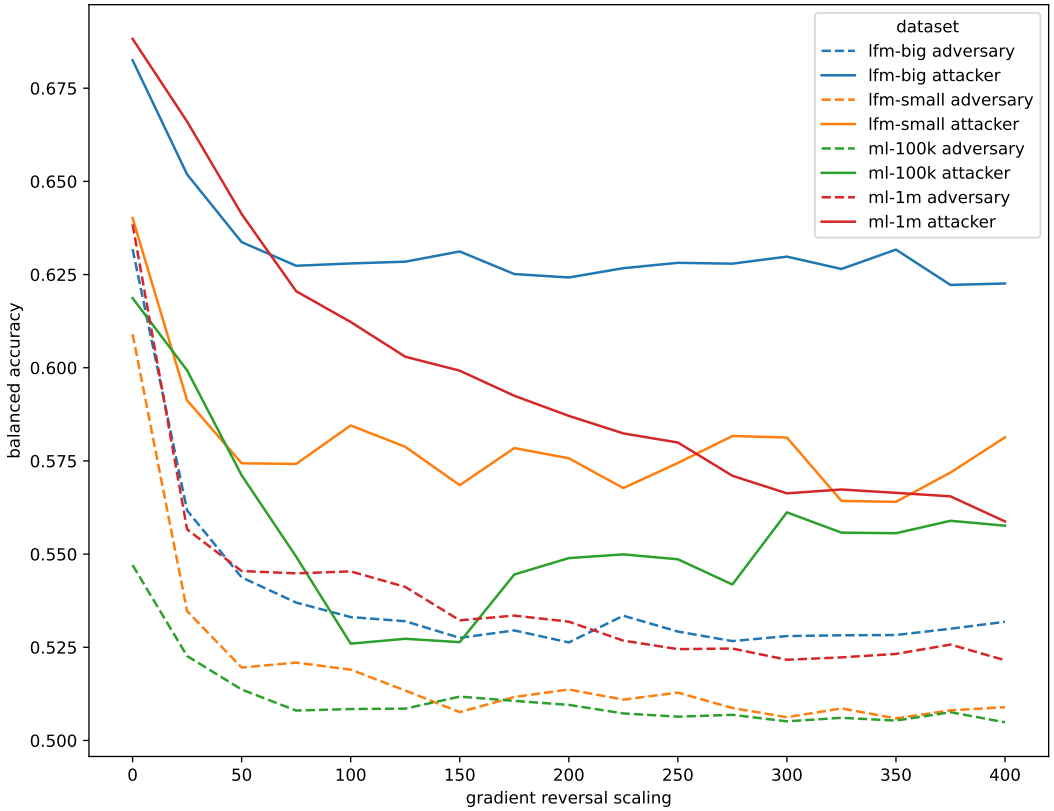


Figure 5.4.: Balanced accuracy comparison of adversaries and attackers on the test set for varying gradient reversal scaling λ of ADV-MULTVAE.

Dataset	Model	Gradient Scaling	Adv. BAcc	Atk. BAcc
ML-100K	MULTVAE	0	0.547	0.619
	ADV-MULTVAE	350	0.505	0.556
ML-1M	MULTVAE	0	0.638	0.688
	ADV-MULTVAE	300	0.522	0.566
LFM-SMALL	MULTVAE	0	0.609	0.640
	ADV-MULTVAE	275	0.509	0.582
LFM-BIG	MULTVAE	0	0.632	0.683
	ADV-MULTVAE	200	0.526	0.624

Table 5.2.: Comparison of adversarial results between MULTVAE and ADV-MULTVAE evaluated on the test set.

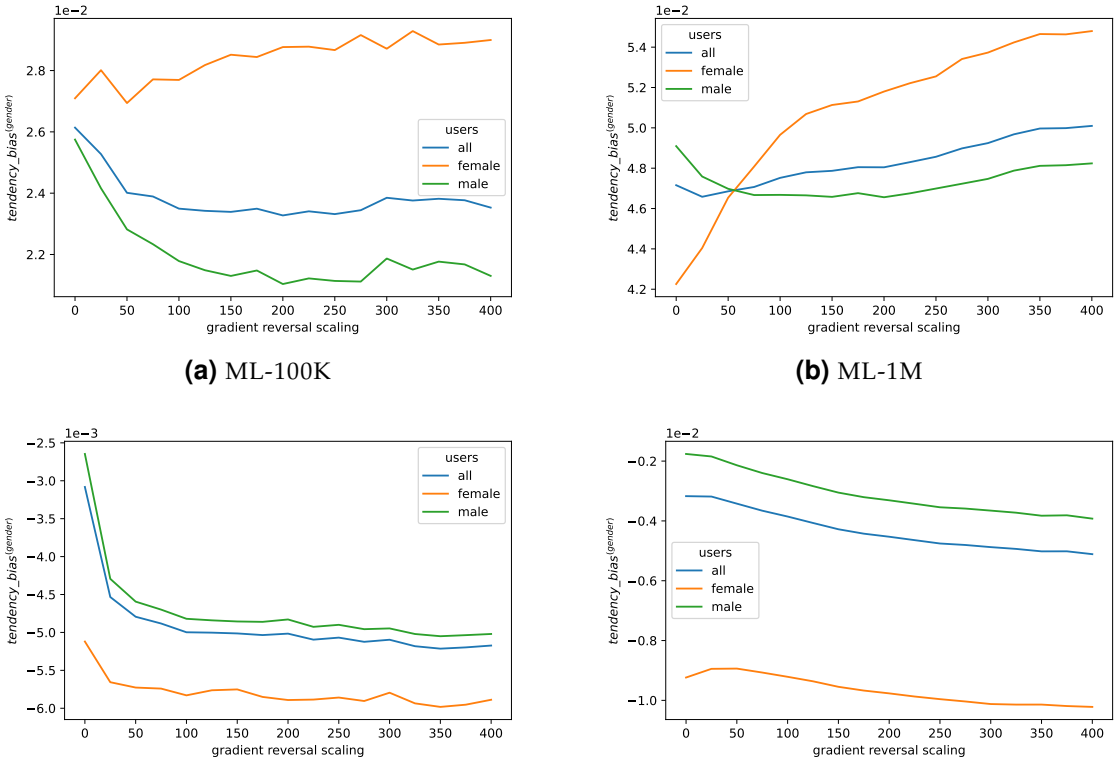
suffers from a lower decrease of RS utility while it more strongly mitigates the encoding of sensitive user information.

To conclude, we compare the adversary and attacker performances in Figure 5.4. As already mentioned above, the attackers may be able to retrieve much more sensitive user information than adversaries. It is especially prominent for the LFM-BIG dataset, where the BAcc of adversaries is 0.526 at $\lambda = 200$, meanwhile, the BAcc of the attacker is 0.632, approx. 0.10 higher than the adversaries'. Nonetheless, adversarial training manages to mitigate the encoding of sensitive information by a lot, as it reduces the attackers BAcc's by up to 0.12.

We provide an overview of the selected adversarial scaling values, based on the adversarial performances on the validation set, in Table 5.2. Moreover, we also list the values of the aforementioned metrics evaluated on the test set.

5.3. User Tendency Bias

Figure 5.5 reports the $tendency\_bias^{(gender)}$ of different model configurations (w. r. t. λ), i. e., the difference of a RS's perceived to actual users ground truths tendencies. As we can see, MULTVAE ($\lambda = 0$) is by default already exhibits biases as it recommending items inclined to a certain user group: male users for ML-100K and ML-1M; female users for



(a) ML-100K **(b)** ML-1M

(c) LFM-SMALL **(d)** LFM-BIG

Figure 5.5.: Comparison of tendency biases different model configurations w. r. t. gradient reversal scaling λ exhibit on different datasets.

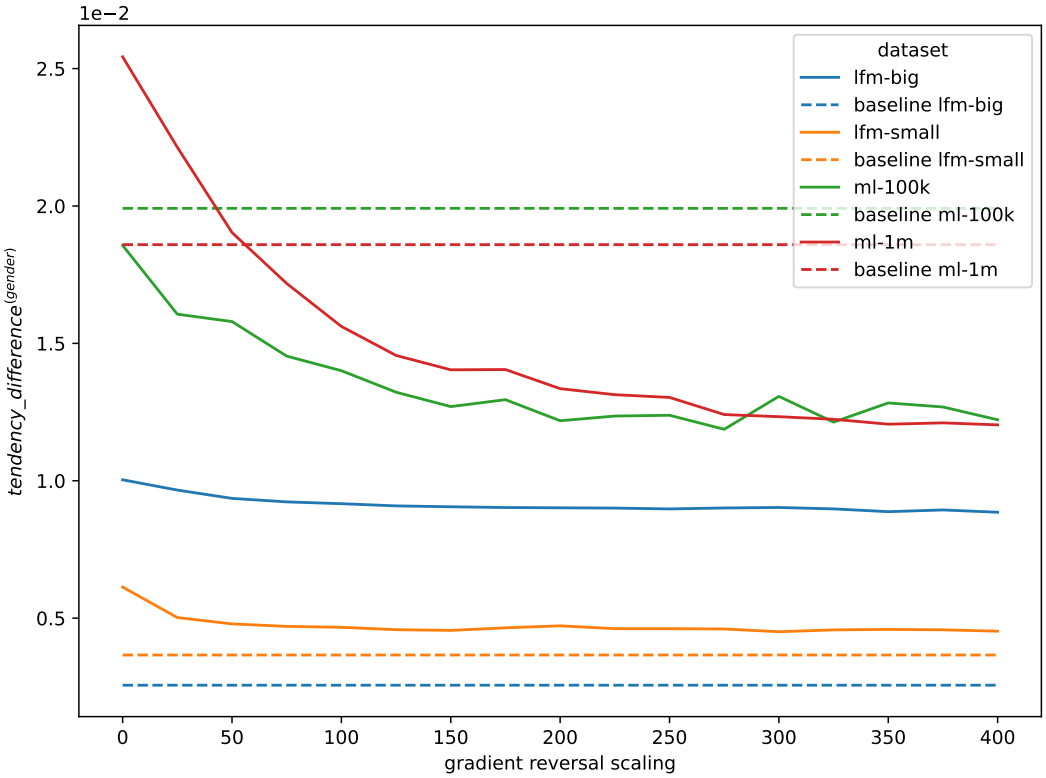
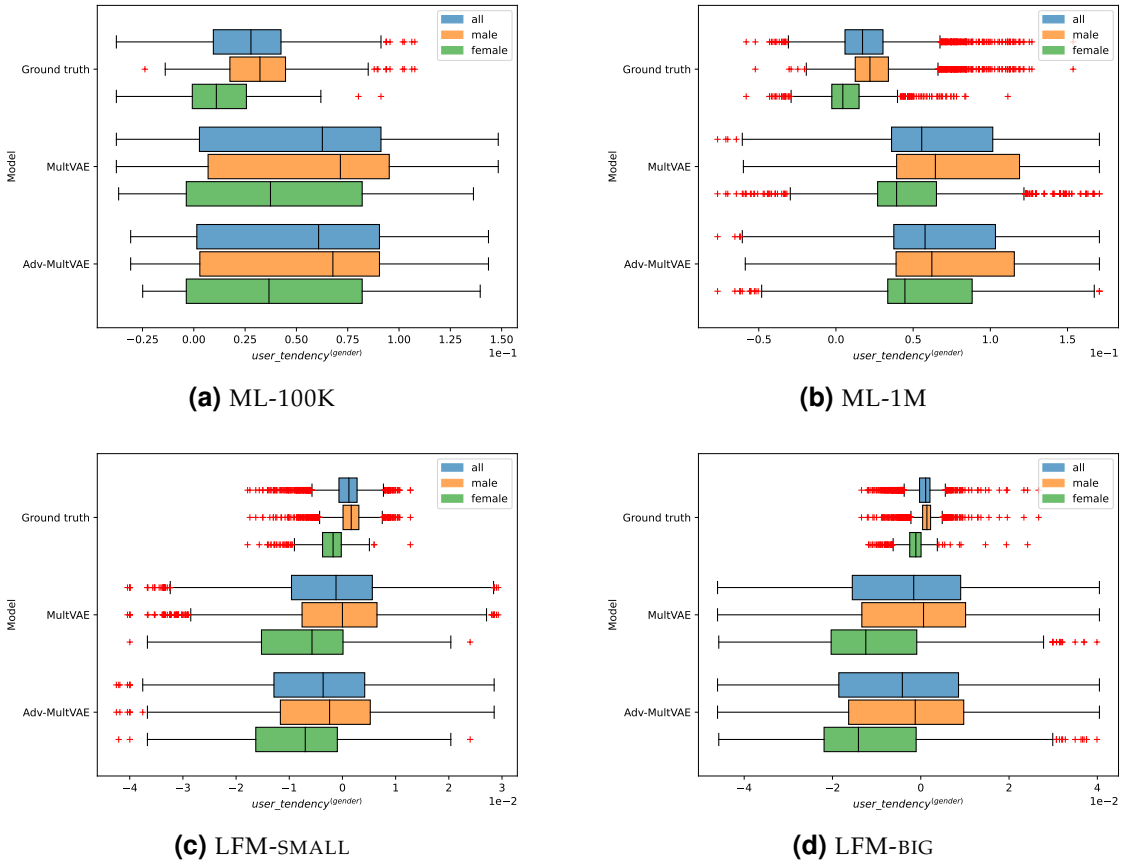


Figure 5.6.: Differences of gender tendencies of recommended items between male and female users. The dashed lines indicate the baseline differences.

LFM-SMALL and LFM-BIG. In general, $tendency\_bias^{(gender)}$ values of high magnitude indicate worse performance, as recommendations do not reflect the users actual interests anymore. Moreover, for increasing gradient reversal scaling λ, the model bias over all users as well as the individual user groups increases further for the LFM-SMALL (c) and LFM-BIG (d) datasets. For ML-100K (a), the overall shift is toward neutral content. In detail, while female users experience a shift toward more male-preferred content, while male users experience a large shift toward more neutral content, leading to overall more neutrality of recommendations. For ML-1M (b), female users experience a huge shift toward male-preferred content, while male users receive to more neutral items. This leads to recommendations of more male preferred items. Not only does the ADV-MULTVAE maintain $tendency\_bias^{(gender)}$, it also enhances it for all but the ML-100K dataset.



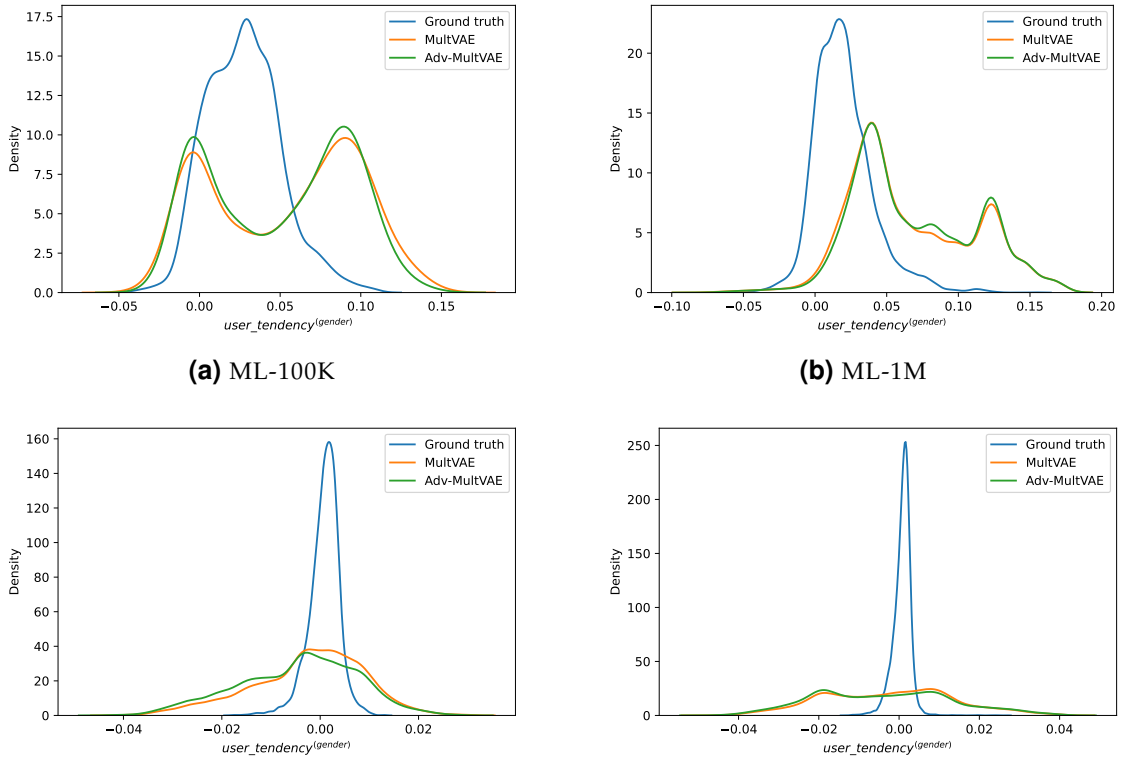
(a) ML-100K **(b)** ML-1M

(c) LFM-SMALL **(d)** LFM-BIG

Figure 5.7.: Users' user tendencies of ground truth (interacted items) and $top-$k recommended items from MULTVAE and ADV-MULTVAE on different datasets.

Therefore, until now, only negative influences of adversarial training could be observed. So is there no positive effect using ADV-MULTVAE? Figure 5.6 reports the tendency differences between male and female users. We can observe a clear trend: adversarial training (with increasing λ) reduces the difference of recommended items between the user groups. For the ML datasets, this trend leads to differences for the user groups much smaller than the users previously expressed. On the other hand, the differences for the LFM datasets is much bigger than the baselines', such that they hardly approach them with increasing λ.

Figure 5.7 provides an alternative, in detail view for the observations above. Here, the wider boxes indicate that the range of recommended user tendencies increases, and thus



(a) ML-100K **(b)** ML-1M

(c) LFM-SMALL **(d)** LFM-BIG

Figure 5.8.: Users' user tendency distributions of ground truth (interacted items) and top-*k* recommended items from MULTVAE and ADV-MULTVAE on different datasets.

lead to drastically more item tendency diversity. Moreover, there are far less outliers visible compared to the ground truth. The previously described shift in recommended items is again visible by checking the vertical median bar inside the boxes. It is worth to highlight that recommender systems support subgroups of users with unusually strong preferences, as their recommended tendencies shift toward the side with the most outliers.

Additionally, Figure 5.8 also visualizes the shift toward recommending items that are more popular in the female population for the LFM-SMALL (c) and LFM-BIG (d) datasets, and a shift toward more popular items in the male population for the ML-100K (a) and ML-1M (b) datasets. Moreover, the tails of the ground truth distribution are also enhanced, again indicating that recommender systems support outlying users with stronger item tendencies.

6. Conclusion

This work addresses the challenge of mitigating societal biases in RSs from the user perspective. To this end, we extend the widely-used MULTVAE model with an adversarial component, and propose the novel ADV-MULTVAE architecture. Morevoer, the introduction of two new bias metrics (*tendency_bias*$^{(a)}$ and *tendency_difference*$^{(a)}$) enables further insight into the behaviour of recommender systems in general, and the effects of adversarial training.

Our approach aims to decrease the model bias in terms of latent information about the protected user attribute in the model and consequently also in the provided recommendations. We conduct experiments on two datasets (ML-1M and LFM2B-DB) and evaluate the amount of recoverable sensitive user information (gender in our experiments) from the models, our defined bias metrics, as well as the models' recommendation accuracy. Our results show that the introduced ADV-MULTVAE model provides a substantial reduction in the amount of encoded protected information, offering a bias- and privacy-aware alternative. Due to a performance decrease in comparison to MULTVAE, there is a trade-off between bias and model utility. The more similar recommendations for male and female users from ADV-MULTVAE may be beneficial for certain application domains (e. g., job recommendations), while perhaps undesired in others (e. g., multimedia consumption).

We envision addressing the shift of recommendations toward the preferences of some user group as a future direction of this work. Analyzing how end-users perceive these changes regarding the biases may be another valuable research question. Moreover, finding a balance between BAcc and nDCG for optimization and model selection might be a possibility to minimize the slight performance loss of ADV-MULTVAE.

JOHANNES KEPLER
UNIVERSITY LINZ

Bibliography

[1] Christian Ganhör, David Penz, Navid Rekabsaz, Oleg Lesota, and Markus Schedl. "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 '22. Madrid, Spain: Association for Computing Machinery, 2022, pp. 2142–2147. ISBN: 9781450387323. DOI: 10.1145/3477495.3531820. URL: https://doi.org/10.1145/3477495.3531820 (cited on page i).

[2] Alessandro B. Melchiorre, Navid Rekabsaz, Emilia Parada-Cabaleiro, Stefan Brandl, Oleg Lesota, and Markus Schedl. "Investigating gender fairness of recommendation algorithms in the music domain". In: *Inf. Process. Manag.* 58.5 (2021), p. 102666. DOI: 10.1016/j.ipm.2021.102666. URL: https://doi.org/10.1016/j.ipm.2021.102666 (cited on pages 1, 2, 11–13).

[3] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. "Variational Autoencoders for Collaborative Filtering". In: *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*. Ed. by Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis. ACM, 2018, pp. 689–698. DOI: 10.1145/3178876.3186150. URL: https://doi.org/10.1145/3178876.3186150 (cited on pages 1, 5).

[4] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. "Are we really making much progress? A worrying analysis of recent neural recommendation approaches". In: *Proceedings of the 13th ACM Conference on Recommender Systems, RecSys 2019, Copenhagen, Denmark, September 16-20, 2019*. Ed. by Toine Bogers, Alan Said, Peter Brusilovsky, and Domonkos Tikk. ACM, 2019, pp. 101–109. DOI: 10.1145/3298689.3347058. URL: https://doi.org/10.1145/3298689.3347058 (cited on pages 1, 2).

[5] Ningxia Wang and Li Chen. "User Bias in Beyond-Accuracy Measurement of Recommendation Algorithms". In: *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*. Ed. by Humberto Jesús Corona Pampín, Martha A. Larson, Martijn C. Willemsen, Joseph A. Konstan, Julian J. McAuley, Jean Garcia-Gathright, Bouke Huurnink, and Even Oldridge. ACM, 2021, pp. 133–142. DOI: 10.1145/3460231.3474244. URL: https://doi.org/10.1145/3460231.3474244 (cited on page 2).

[6] Michael D. Ekstrand, Mucun Tian, Mohammed R. Imran Kazi, Hoda Mehrpouyan, and Daniel Kluver. "Exploring author gender in book rating and recommendation". In: *Proceedings of the 12th ACM Conference on Recommender Systems, RecSys 2018, Vancouver, BC, Canada, October 2-7, 2018*. Ed. by Sole Pera, Michael D. Ekstrand, Xavier Amatriain, and John O'Donovan. ACM, 2018, pp. 242–250. DOI: 10.1145/3240323.3240373. URL: https://doi.org/10.1145/3240323.3240373 (cited on page 2).

[7] Mehdi Elahi, Dietmar Jannach, Lars Skjærven, Erik Knudsen, Helle Sjøvaag, Kristian Tolonen, Øyvind Holmstad, Igor Pipkin, Eivind Throndsen, Agnes Stenbom, Eivind Fiskerud, Adrian Oesch, Loek Vredenberg, and Christoph Trattner. "Towards responsible media recommendation". In: *AI and Ethics* (Nov. 2021). ISSN: 2730-5961. DOI: 10.1007/s43681-021-00107-7. URL: https://doi.org/10.1007/s43681-021-00107-7 (cited on page 2).

[8] Christine Bauer. "Allowing for equal opportunities for artists in music recommendation". In: *CoRR* abs/1911.05395 (2019). arXiv: 1911.05395. URL: http://arxiv.org/abs/1911.05395 (cited on page 2).

[9] Lisette Espín-Noboa, Claudia Wagner, Markus Strohmaier, and Fariba Karimi. "Inequality and Inequity in Network-based Ranking and Recommendation Algorithms". In: *CoRR* abs/2110.00072 (2021). arXiv: 2110.00072. URL: https://arxiv.org/abs/2110.00072 (cited on page 2).

[10] Navid Rekabsaz, Simone Kopeinik, and Markus Schedl. "Societal Biases in Retrieved Contents: Measurement Framework and Adversarial Mitigation of BERT Rankers". In: *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*. Ed. by Fernando Diaz, Chirag Shah, Torsten Suel, Pablo Castells, Rosie Jones, and Tet-

suya Sakai. ACM, 2021, pp. 306–316. DOI: 10.1145/3404835.3462949. URL: https://doi.org/10.1145/3404835.3462949 (cited on pages 2, 3, 7).

[11] Navid Rekabsaz and Markus Schedl. "Do Neural Ranking Models Intensify Gender Bias?" In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*. Ed. by Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu. ACM, 2020, pp. 2065–2068. DOI: 10.1145/3397271.3401280. URL: https://doi.org/10.1145/3397271.3401280 (cited on page 2).

[12] Ghazaleh Beigi, Ahmadreza Mosallanezhad, Ruocheng Guo, Hamidreza Alvari, Alexander Nou, and Huan Liu. "Privacy-Aware Recommendation with Private-Attribute Protection using Adversarial Learning". In: *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*. Ed. by James Caverlee, Xia (Ben) Hu, Mounia Lalmas, and Wei Wang. ACM, 2020, pp. 34–42. DOI: 10.1145/3336191.3371832. URL: https://doi.org/10.1145/3336191.3371832 (cited on pages 2, 3, 19).

[13] Ghazaleh Beigi, Kai Shu, Ruocheng Guo, Suhang Wang, and Huan Liu. "Privacy Preserving Text Representation Learning". In: *Proceedings of the 30th ACM Conference on Hypertext and Social Media, HT 2019, Hof, Germany, September 17-20, 2019*. Ed. by Claus Atzenbeck, Jessica Rubart, and David E. Millard. ACM, 2019, pp. 275–276. DOI: 10.1145/3342220.3344925. URL: https://doi.org/10.1145/3342220.3344925 (cited on page 2).

[14] Udi Weinsberg, Smriti Bhagat, Stratis Ioannidis, and Nina Taft. "BlurMe: inferring and obfuscating user gender based on ratings". In: *Sixth ACM Conference on Recommender Systems, RecSys '12, Dublin, Ireland, September 9-13, 2012*. Ed. by Padraig Cunningham, Neil J. Hurley, Ido Guy, and Sarabjot Singh Anand. ACM, 2012, pp. 195–202. DOI: 10.1145/2365952.2365989. URL: https://doi.org/10.1145/2365952.2365989 (cited on page 2).

[15] F. Maxwell Harper and Joseph A. Konstan. "The MovieLens Datasets: History and Context". In: *ACM Trans. Interact. Intell. Syst.* 5.4 (Dec. 2015). ISSN: 2160-6455. DOI: 10.1145/2827872. URL: https://doi.org/10.1145/2827872 (cited on pages 2, 11).

[16] Markus Schedl, Stefan Brandl, Oleg Lesota, Emilia Parada-Cabaleiro, David Penz, and Navid Rekabsaz. "LFM-2b: A Dataset of Enriched Music Listening Events for Recommender Systems Research and Fairness Analysis". In: *ACM SIGIR Conference on Human Information Interaction and Retrieval*. 2022, pp. 337–341 (cited on page 2).

[17] Yashar Deldjoo, Tommaso Di Noia, and Felice Antonio Merra. "A Survey on Adversarial Recommender Systems: From Attack/Defense Strategies to Generative Adversarial Networks". In: *ACM Comput. Surv.* 54.2 (2021), 35:1–35:38. DOI: 10.1145/3439729. URL: https://doi.org/10.1145/3439729 (cited on page 3).

[18] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. "Mitigating Unwanted Biases with Adversarial Learning". In: *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. New Orleans LA USA: ACM, Dec. 2018, pp. 335–340. ISBN: 978-1-4503-6012-8. DOI: 10.1145/3278721.3278779 (cited on page 3).

[19] Le Wu, Lei Chen, Pengyang Shao, Richang Hong, Xiting Wang, and Meng Wang. "Learning Fair Representations for Recommendation: A Graph-based Perspective". In: *Proceedings of the Web Conference 2021*. Ljubljana Slovenia: ACM, Apr. 2021, pp. 2198–2208. ISBN: 978-1-4503-8312-7. DOI: 10.1145/3442381.3450015 (cited on page 3).

[20] Yunqi Li, Hanxiong Chen, Shuyuan Xu, Yingqiang Ge, and Yongfeng Zhang. "Towards Personalized Fairness Based on Causal Notion". In: *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. Virtual Event Canada: ACM, July 2021, pp. 1054–1063. ISBN: 978-1-4503-8037-9. DOI: 10.1145/3404835.3462966 (cited on page 3).

[21] Ziwei Zhu, Jianling Wang, and James Caverlee. "Measuring and Mitigating Item Under-Recommendation Bias in Personalized Ranking Systems". In: *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*. Ed. by Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu. ACM, 2020, pp. 449–458. DOI: 10.1145/3397271.3401177. URL: https://doi.org/10.1145/3397271.3401177 (cited on page 3).

[22] Chuhan Wu, Fangzhao Wu, Xiting Wang, Yongfeng Huang, and Xing Xie. "Fairness-aware News Recommendation with Decomposed Adversarial Learning". In: vol. 35. 5. May 2021, pp. 4462–4469. DOI: 10.1609/aaai.v35i5.16573. URL: https://ojs.aaai.org/index.php/AAAI/article/view/16573 (cited on page 3).

[23] George Zerveas, Navid Rekabsaz, Daniel Cohen, and Carsten Eickhoff. "Mitigating bias in search results through set-based document reranking and neutrality regularization". In: *Proceedings of the 45th International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2022*. ACM, 2022 (cited on page 3).

[24] Klara Krieg, Emilia Parada-Cabaleiro, Markus Schedl, and Navid Rekabsaz. "Do Perceived Gender Biases in Retrieval Results Affect Relevance Judgements?" In: *Proceedings of the Workshop on Algorithmic Bias in Search and Recommendation at the European Conference on Information Retrieval (ECIR-BIAS 2022)*. 2022 (cited on page 3).

[25] Klara Krieg, Emilia Parada-Cabaleiro, Gertraud Medicus, Oleg Lesota, Markus Schedl, and Navid Rekabsaz. "Grep-BiasIR: A Dataset for Investigating Gender Representation-Bias in Information Retrieval Results". In: *arXiv preprint arXiv:2201.07754* (2022) (cited on page 3).

[26] Jiawei Chen, Hande Dong, Xiang Wang, Fuli Feng, Meng Wang, and Xiangnan He. "Bias and Debias in Recommender System: A Survey and Future Directions". In: *arXiv:2010.03240 [cs]* (Oct. 2020). arXiv: 2010.03240 [cs] (cited on page 4).

[27] Sirui Yao and Bert Huang. *Beyond Parity: Fairness Objectives for Collaborative Filtering*. Nov. 2017. arXiv: 1705.08804 [cs, stat] (cited on page 4).

[28] Oleg Lesota, Alessandro B. Melchiorre, Navid Rekabsaz, Stefan Brandl, Dominik Kowald, Elisabeth Lex, and Markus Schedl. "Analyzing Item Popularity Bias of Music Recommender Systems: Are Different Genders Equally Affected?" In: *RecSys '21: Fifteenth ACM Conference on Recommender Systems, Amsterdam, The Netherlands, 27 September 2021 - 1 October 2021*. Ed. by Humberto Jesús Corona Pampín, Martha A. Larson, Martijn C. Willemsen, Joseph A. Konstan, Julian J. McAuley, Jean Garcia-Gathright, Bouke Huurnink, and Even Oldridge. ACM, 2021, pp. 601–606. DOI: 10.1145/3460231.3478843. URL: https://doi.org/10.1145/3460231.3478843 (cited on pages 4, 18).

[29] Diederik P. Kingma and Max Welling. "Auto-Encoding Variational Bayes". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2014. URL: http://arxiv.org/abs/1312.6114 (cited on pages 4, 5).

[30] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. Ed. by Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger. 2014, pp. 2672–2680. URL: https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html (cited on page 7).

[31] Yanai Elazar and Yoav Goldberg. "Adversarial Removal of Demographic Attributes from Text Data". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*. Ed. by Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii. Association for Computational Linguistics, 2018, pp. 11–21. DOI: 10.18653/v1/d18-1002. URL: https://doi.org/10.18653/v1/d18-1002 (cited on page 7).

[32] Qizhe Xie, Zihang Dai, Yulun Du, Eduard H. Hovy, and Graham Neubig. "Controllable Invariance through Adversarial Feature Learning". In: *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*. Ed. by Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett. 2017, pp. 585–596. URL: https://proceedings.neurips.cc/paper/2017/hash/8cb22bdd0b7ba1ab13d742e22eed8da2-Abstract.html (cited on page 7).

[33] Yaroslav Ganin and Victor S. Lempitsky. "Unsupervised Domain Adaptation by Backpropagation". In: *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. Ed. by Francis R. Bach and David M. Blei. Vol. 37. JMLR Workshop and Conference Proceedings. JMLR.org, 2015, pp. 1180–1189. URL: http://proceedings.mlr.press/v37/ganin15.html (cited on page 7).

[34] Tal Laor and Yair Galily. "Who'S Clicking on on-Demand? Media Consumption Patterns of Generations Y & Z". In: *Technology in Society* 70 (Aug. 2022), p. 102016. ISSN: 0160791X. DOI: 10.1016/j.techsoc.2022.102016 (cited on page 9).

[35] Zaiqiao Meng, Richard McCreadie, Craig Macdonald, and Iadh Ounis. "Exploring Data Splitting Strategies for the Evaluation of Recommendation Models". In: *Rec-*

Sys 2020: Fourteenth ACM Conference on Recommender Systems, Virtual Event, Brazil, September 22-26, 2020. Ed. by Rodrygo L. T. Santos, Leandro Balby Marinho, Elizabeth M. Daly, Li Chen, Kim Falk, Noam Koenigstein, and Edleno Silva de Moura. ACM, 2020, pp. 681–686. DOI: 10.1145/3383313.3418479. URL: https://doi.org/10.1145/3383313.3418479 (cited on page 12).

[36] Kay Henning Brodersen, Cheng Soon Ong, Klaas Enno Stephan, and Joachim M. Buhmann. "The Balanced Accuracy and Its Posterior Distribution". In: *20th International Conference on Pattern Recognition, ICPR 2010, Istanbul, Turkey, 23-26 August 2010*. IEEE Computer Society, 2010, pp. 3121–3124. DOI: 10.1109/ICPR.2010.764. URL: https://doi.org/10.1109/ICPR.2010.764 (cited on page 13).

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. "PyTorch: An Imperative Style, High-Performance Deep Learning Library". In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf (cited on page 14).

JOHANNES KEPLER
UNIVERSITY LINZ

A. Hyperparameter settings

		ML-100K	ML-1M	LFM-SMALL	LFM-BIG
model	size	[5] [10] [20] [50] **[100]** [200] [50, 10] [100, 10] [200, 20]	[10] [20] [50] **[100]** [200] [100, 20] [100, 50] [200, 20] [200, 50] [200, 100]	[10] [20] [50] [100] [200] **[500]** [700] [100, 20] [200, 20]	[50] [100] [200] [300] **[500]** [700] [100, 20] [200, 20] [500, 20] [500, 100]
	input dropout	0.0 **0.1** 0.3 0.5 0.7		0.0 0.1 **0.3** 0.5 0.7	
	batch size	64			
	beta	0, 0.4, **0.8**, 1, 1.2			
	activation function	**ReLU**			
optimizer	learning rate	**1e-3** 5e-4 1e-4 5e-5		1e-3 **5e-4** 1e-4 5e-5	
	weight decay	0 5e-4 **1e-4** 5e-4 1e-3 5e-3			

Table A.1.: Hyperparameters used for search of best configuration for MULTVAE. Final values are marked in **bold**.

		ML-100K	ML-1M	LFM-SMALL	LFM-BIG
adversary	size	[2] **[5, 2]** [10, 2] [20, 2] [50, 2]			
	dropout	0.0 0.1 0.3 0.5 **0.7**		0.0 0.1 0.3 **0.5** 0.7	
	loss weight	0.5, **1**, 1.5, 2, 5, 10			
	number of adversaries	1, 2, 5, 7, **10**			
	gradient scaling	0 - 400 (**350**)	0 - 400 (**300**)	0 - 400 (**275**)	0 - 400 (**200**)
	activation function	**ReLU**			
optimizer	learning rate	**1e-3** 5e-4 1e-4 5e-5			
	weight decay	0 5e-4 **1e-4** 5e-4 1e-3 5e-3			
LR scheduler	reduction factor	**0.1**			
	min LR	5e-5 1e-5 5e-6 **1e-6**			
	patience	3, **5**, 7			

Table A.2.: Hyperparameters used for search of best configuration for ADV-MULTVAE. Final values are marked in **bold**.