# Towards Debiasing DNN Models from Spurious Feature Influence

**Mengnan Du**[1], **Ruixiang Tang**[2], **Weijie Fu**[3], **Xia Hu**[2]
[1]Texas A&M University
[2]Rice University
[3]Hefei University of Technology
dumengnan@tamu.edu, {rt39,xia.hu}@rice.edu, fwj.edu@gmail.com

## Abstract

Recent studies indicate that deep neural networks (DNNs) are prone to show discrimination towards certain demographic groups. We observe that algorithmic discrimination can be explained by the high reliance of the models on fairness sensitive features. Motivated by this observation, we propose to achieve fairness by suppressing the DNN models from capturing the spurious correlation between those fairness sensitive features with the underlying task. Specifically, we first train a bias-only teacher model which is explicitly encouraged to maximally employ fairness sensitive features for prediction. The teacher model then counter-teaches a debiased student model so that the interpretation of the student model is orthogonal to the interpretation of the teacher model. The key idea is that since the teacher model relies explicitly on fairness sensitive features for prediction, the orthogonal interpretation loss enforces the student network to reduce its reliance on sensitive features and instead capture more task-relevant features for prediction. Experimental analysis indicates that our framework substantially reduces the model's attention on fairness sensitive features. Experimental results on four datasets further validate that our framework has consistently improved model fairness with respect to group fairness metrics, with a comparable or even better accuracy.

## Introduction

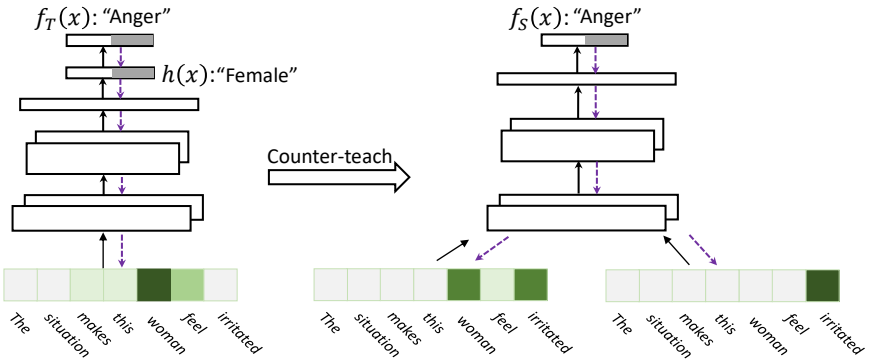
DNN models are increasingly being used in high-stake decision making applications that affect individuals. However, these models might exhibit algorithmic bias behaviors (Nagpal et al. 2019; Du et al. 2020; Kiritchenko and Mohammad 2018; Wan et al. 2021). Specifically, DNN models place certain privileged groups at a systematic advantage and exhibit discrimination with respect to certain unprivileged groups. For example, a recruiting tool believes that males are more qualified and gives much lower ratings to females (Kiritchenko and Mohammad 2018), loan eligibility system negatively rates African Americans, and the recidivism prediction system predicts that African American inmates are three times more likely to be classified as 'high risk' than European American inmates (Angwin et al. 2016), to name a few. Many algorithmic discriminations are not justified, and the bias problem might cause adverse impacts on individuals and society. Therefore, designing mitigation methods to reduce the algorithmic bias of DNN models has attracted increasing attention recently (Mehrabi et al. 2022; Wu et al. 2021; Tang et al. 2021).



Our work is motivated by the observation that the bias behavior of standard DNN models is a direct result of their high reliance on fairness sensitive features in inputs. Here fairness sensitive features denote features (e.g., ZIP code and surname) that are highly predictive of certain protected attribute (e.g., race). As a result, the underlying prediction task (e.g., mortgage application) would highly rely on the protected attributes (e.g., race) for prediction and introduce discrimination for certain groups (e.g., African Americans). Based on this observation, we propose to mitigate bias by suppressing the model from capturing superficial correlations of fairness sensitive features with the prediction task, while forcing it to concentrate on task-relevant features.

Decorrelating fairness sensitive features with class labels for DNN models is a technically challenging problem. Firstly, one challenge lies in how to locate fairness sensitive features in input. One straightforward idea is to label the whole training set by domain experts or crowd workers. This would lead to suboptimal results. On one hand, crowd sourcing labelling is too time consuming and the labelling quality is not guaranteed (McDonnell et al. 2016). On the other hand, many seemingly innocuous features may be highly correlated with protected attributes and cause model bias. This makes it extremely hard to annotate an exhaustive list of sensitive features. Secondly, it is also challenging to utilize the sensitive features even if we could obtain such labels. A straightforward way is to delete these features, which however is impractical in many applications.

To address these challenges, we propose a general framework, called DeFI (Decorrelating Feature Influence), to discorrelate the main prediction task and fairness sensitive features for bias mitigation. We introduce a bias-only teacher network that primarily leverages sensitive features in the input to make predictions. Fairness sensitive features can be automatically localized by the biased teacher network. This teacher network then counter-teaches a debiased student network, so that the interpretation of the student model is orthogonal to the interpretation of the teacher model. The key idea is that since the teacher model relies explicitly on fairness sensitive features for prediction, the orthogonal interpretation loss enforces the student network to focus more




(a) Teacher DNN (b) Student without teaching (C) Student after teaching

Figure 1: An illustrative example of the proposed DeFI framework, where the task is for sentiment classification and the protected attribute is gender. (a) The bias-only teacher model mostly relies on fairness sensitive feature, i.e., '*woman*' for prediction. (b) Without teaching, the student DNN will pick up both undesirable fairness sensitive features, i.e., '*woman*', and features reflective of sentiment, i.e., '*irritated*'. (c) After counter-teaching from the teacher network, the student DNN will exclusively concentrate on task-relevant features, i.e., '*irritated*', for prediction.

on task-relevant features for prediction. At test time, our method does not need access to protected attributes, since collecting protected attributes is often not allowed in real-world applications. The major contributions of this paper are summarized as follows:

- We propose a general bias mitigation framework, called DeFI, which could reduce model discrimination via decorrelating the prediction task and fairness sensitive features.
- DeFI is applicable to different DNN architectures and can be easily extended to tackle multiple protected attributes (e.g., race and gender) to achieve compositional fairness.
- Experimental results show that DeFI could increase the performance with respect to demographic parity and equality of odds metrics, while maintaining the original prediction accuracy. The analysis further indicates that DeFI has reduced attention on fairness sensitive features.

## The Proposed Framework

In this section, we introduce the proposed DeFI framework (Decorrelating Feature Influence). We formulate it into a two-step procedure: 1) first training a biased teacher network which deliberately maximizes the usage of fairness sensitive features for prediction, 2) then training a student network where its interpretation is orthogonal to the interpretation of the teacher network. We design the teaching in two ways: explicit decorrelation and implicit decorrelation.

### Problem Statement

We first introduce the notations as well as fairness measurements. Then we present feature influence analysis that serves as the basic motivation for our proposed mitigation method.

**Notations.** Consider a classification problem with labeled examples: $\{x, y, a\} \sim p_{data}$, where $x \in \mathcal{X}$ is input feature, and $y \in \mathcal{Y}$ is the label that we want to predict. Besides, $a \in \mathcal{A} = \{0, ..., K\}$ is a $K$ categorical *protected attribute* annotation, such as race, gender, and age, where there exist certain unprivileged and privileged groups. We assume that the protected attributes $\mathcal{A}$ can only be used during the training phase and are not accessible during the inference time (post-training). Our goal here is to learn a classification model $\hat{y} = f(x)$ which is predictive of label $y$, while at the same time satisfying certain fairness measurements with regard to a protected attribute $a$. In this work, we restrict our attention to models that make a binary classification decision, i.e., $\mathcal{Y} = \{0, 1\}$, where 1 and 0 denote favorable outcome and unfavorable outcome, respectively.

**Fairness Measurements.** We use three statistic (group) fairness metrics to assess the fairness of the model (Gajane and Pechenizkiy 2018). The *demographic parity* metric (Feldman et al. 2015) is defined as the probability ratio of favorable outcome between unprivileged group ($a = 0$) and privileged group ($a = 1$): $\mathcal{F}_{parity} = \frac{p(\hat{y}=1|a=0)}{p(\hat{y}=1|a=1)}$, where $\hat{y}$ is the model prediction and 1 denotes the favorable outcome. The *equality of opportunity* metric (Hardt et al. 2016; Zafar et al. 2017) is defined as the true positive rate difference between unprivileged group and privileged group: $\mathcal{F}_{opty} = p(\hat{y} = 1|a = 0, y = 1) - p(\hat{y} = 1|a = 1, y = 1)$. *Equality of odds* metric (Hardt et al. 2016) also takes false positive rate into consideration: $\mathcal{F}_{odds} = p(\hat{y} = 1|a = 0, y = 0) - p(\hat{y} = 1|a = 1, y = 0) + \mathcal{F}_{opty}$. Furthermore, we also use *accuracy* $\mathcal{F}_{acc}$ to assess the utility of the model.

**Feature Influence Analysis.** Our work is based on the experimental observation that discrimination is mainly caused by the model's dependence on *fairness sensitive features* for prediction. Here *fairness sensitive features* are subset of features in the input $x$ that are highly predictive of *protected attribute* $a$. We use an interpretation method (Sundararajan, Taly, and Yan 2017) to analyze the feature importance distribution for different types of features. For a text-based sentiment classification task using EEC dataset (Kiritchenko and Mohammad 2018), the interpretation heatmap indicates that DNN model heavily relies on fairness sensitive features for prediction. An example is illustrated in Fig. 1(b). For this task, the word '*woman*' is fairness sen-

sitive feature, which is highly correlated with protected attribute $a$ = Female. The model pays comparable attention to word '*woman*' with '*irritated*', indicating that it has associated females with the negative anger sentiment. Due to the data distribution imbalance in the training set, fairness sensitive features could have high correlation with certain class labels. Most current DNNs follow the data-driven learning paradigm. The trained models would capture *superficial correlation* between fairness sensitive features and the label, amplifying these biases and taking a *shortcut* to make predictions (Geirhos et al. 2021). Eventually, the DNN models show discrimination towards certain demographic groups.

## Decorrelating Feature Influence (DeFI)

Based on the analysis in the last section, we propose to achieve fairness by decorrelating feature influence from fairness sensitive features to the prediction label (see Fig. 1). However, *it is challenging to locate the fairness sensitive features in the input*. Thus, we formulate the decorrelation into the knowledge distillation framework (Hinton, Vinyals, and Dean 2015; Phuong and Lampert 2019), while through counter-teaching. We construct a bias-only teacher network which is trained to maximally utilize fairness sensitive features for prediction. Then the teacher network is further employed to counter-teach a debiased student network.

**Constructing a Bias-Only Teacher Network.** Our hypothesis is that the input contains fairness sensitive features and task-relevant features, and our goal is to separate them automatically. Specifically, we build a bias-only teacher model which maximally utilizes the fairness sensitive features for prediction. The teacher is denoted as $f_T(x) = c(h(x))$, where $h(x)$ is the intermediate representation for input $x$, and $c(\cdot)$ is responsible for mapping the intermediate representation to the final prediction. Note that $h(x)$ only contains $|\mathcal{A}|$ dimensions. The key motivation of using the $|\mathcal{A}|$-dimension input representation $h(x)$ is to force the teacher network to only utilize biased information, i.e., fairness sensitive features in input, to obtain prediction $f_T(x)$.

A two-stage strategy is used to train the bias-only teacher model $f_T(x)$. Firstly, we use the input and the protected attribute annotation $\{x_i, a_i\}_{i=1}^N$ to train the representation $h(x)$. The purpose is to maximize the bias information captured by the representation $h(x)$. Essentially we treat this as the multiclass classification problem. Take the sample in Fig. 1 (a) for example, the input $x$ is the sentence "the situation makes this woman feel irritated", and the protected attribute $a$ is "female". Secondly, we utilize $\{h(x_i), y_i\}_{i=1}^N$ to train the function $c(\cdot)$ to learn the mapping from $h(x)$ to $f_T(x)$. Ultimately, $f_T(x)$ will maximize the use of the most discriminative sensitive features for prediction.

We illustrate the idea using Fig. 1(a). This is a sentiment classification task, and we consider gender bias. The input representation $h(x)$ contains two dimensions, indicating information for male and female, respectively. The teacher network $f_T(x)$ relies mainly on the fairness sensitive feature *'woman'* for prediction, while at the same time paying nearly no attention to task-relevant feature *'irritated'*.

**Counter-Teaching a Debiased Student Network.** Equipped with the bias-only teacher network $f_T(x)$, we could counter-teach a student network $f_S(x)$ to force the student network to utilize complementary knowledge as the teacher network. We propose two strategies to achieve the teaching, including explicitly decorrelating feature influence and implicitly decorrelating feature influence from fairness sensitive features. Ultimately, we could obtain a debiased DNN model which minimally relies on fairness sensitive features for prediction.

## Explicitly Decorrelating Feature Influence

In this section, we introduce how to counter-teach the student network with the bias-only teacher network for bias mitigation. Some fairness sensitive features in input $x_i$ can be used to predict protected attributes $a_i$ with a high probability (Feldman et al. 2015). The high reliance of these features can cause the discrimination of DNNs. Our goal is to explicitly discourage the model from capturing superficial correlations between fairness sensitive features and labels.

We use local DNN interpretability to obtain the contribution of features towards model prediction (Du, Liu, and Hu 2020). It is achieved by attributing the model's prediction to its input features. The final interpretation is illustrated in the format of feature importance vectors, where a higher value indicates a higher contribution score of that feature to the model prediction. We explicitly regularize the interpretation for the student with the interpretation of the teacher network, and the loss function is given as follows:

$$\mathcal{L}_{\text{EX}}(x) = \frac{1}{N}\sum_{i=1}^{N}\langle I(f_T(x_i), x_i), I(f_S(x_i), x_i)\rangle, \qquad (1)$$

where each $I$ represents the local interpretation vector of $x_i$ for the teacher and student network, respectively. The interpretation vector $I$ has the same length as the input $x_i$, and each element of $I$ denotes how relevant a feature within the input $x_i$ can explain the prediction of the model $f(x_i)$. We encourage a smaller inner product and expect that these two vectors are more different from each other. Considering that the biased teacher gives high attention to word '*woman*' in Fig. 1, then the student network is enforced to give near-zero attention to that word instead.

**Interpretation Algorithm.** To generate interpretations, we use a back-propagation based interpretation method named Integrated Gradient (Sundararajan, Taly, and Yan 2017), as it is a model-agnostic interpretation technique applicable to all models that have differentiable output in terms of inputs. Its key idea is to integrate the gradients of $m$ intermediate samples over the straightline path from baseline $x_{base}$ to input $x_i$, which could be denoted as:

$$I(f(x_i), x_i) = (x_i - x_{base}) \cdot \sum_{k=1}^{m} \frac{\partial f(x_{base} + \frac{k}{m}(x_i - x_{base}))}{\partial x_i} \cdot \frac{1}{m}. \qquad (2)$$

The sensitivity of each feature with respect to the prediction is integrated over the spectrum to give the approximate attribution score for each feature. To calculate each gradient, a target label needs to be specified, where we use the ground truth label $y_i$ of $x_i$.

Note that for text classification applications, each input text is composed of $T$ words: $x_i = \{x_i^t\}_{t=1}^T$, and each word $x_i^t \in R^d$ denotes a word embedding with $d$ dimensions. We first compute gradients of the output prediction with respect to individual entries in word embedding vectors, and use the L2 norm to reduce each vector of the gradients to a single attribution value, representing the contribution of each single word. Also, for different inputs $x_i$, we use the same baseline $x_{baseline}$, and fix it as a zero-value vector for tabular input and as zero word embedding for text input.

## Implicitly Decorrelating Feature Influence

We could also train the debiased student network as an ensemble with the biased teacher network. The key idea is to implicitly encourage the student network to use alternative features in the input. The ensemble of probability output from teacher $f_T(x_i)$ and student $f_S(x_i)$ is given as follows:

$$p(y|x) = \text{softmax}(log(p_S(y|x)) + log(p_T(y|x))), \quad (3)$$

where the first term is what we expect the student network to capture, and the second term denotes what the teacher network has learned. In the ensemble learning (Hinton 2002; He, Zha, and Wang 2019), we fix parameters for the teacher network and only update the parameters for the student network. In the following, we show that the *implicit effect* of the ensemble training of Eq.(3) is to force the student to capture complementary features to the teacher, i.e., the interpretation of the student is orthogonal to interpretation of the teacher.

**Relation to Decorrelating Feature Influence.** Suppose each input feature $x$ could be split into two subsets of features: fairness sensitive features $x_{sens}$ which are highly relevant to protected attribute $a$ and the rest features $x_{task}$ which are more relevant to the main prediction task. We could approximately decompose the model prediction $p(x)$ by applying the Bayes rule as follows:

$$\begin{aligned} p(y|x) &= p(y|x_{sens}, x_{task}) && (4a)\\ &\propto p(y|x_{task})p(x_{sens}|y, x_{task}) && (4b)\\ &\propto p(y|x_{task})p(x_{sens}|y) && (4c)\\ &= p(y|x_{task})\frac{p(y|x_{sens})p(x_{sens})}{p(y)} && (4d)\\ &\propto \underbrace{p(y|x_{task})}_{\text{Student}}\underbrace{p(y|x_{sens})}_{\text{Teacher}}/p(y), && (4e) \end{aligned}$$

where Eq. (4b) is obtained by applying the Bayes rule while conditioning on $x_{task}$. Furthermore, suppose that these two sets of features $x_{sens}$ and $x_{task}$ are conditionally independent given label $y$, we could omit $x_{task}$ from $p(x_{sens}|y, x_{task})$ and obtain Eq. (4c). By further applying the Bayes rule for $p(x_{sens}|y)$, we can obtain Eq. (4d). Also considering that the training set is relatively balanced for the label $y$, we omit it from the Eq. (4e), and obtain the formulation of Eq.(3).

A desirable debiased model will mainly rely on task-relevant features, i.e., $x_{task}$, for prediction. Nevertheless, a model trained with cross entropy loss, i.e., $p(y|x)$, will rely on both $x_{sens}$ and $x_{task}$ for prediction. We cannot directly calculate $p(y|x_{sens})$, which is thus obtained from the bias-only teacher network. Using the ensemble learning of Eq.(3), the student network is enforced to capture complementary features to the teacher, i.e., $p(y|x_{task})$ (Clark, Yatskar et al. 2019). The feature influence of the student network is orthogonal to the feature influence of the teacher network, and thus the student network would shift its attention from fairness sensitive features to task-relevant features.

**Adjusting The Influence of Teacher Network.** Sometimes the teacher network could be strongly biased towards certain predictions. Taking Fig. 1(a) for example, the model could output a strong negative sentiment whenever the input is relevant to females. In the preliminary experiments, we find that if we directly add the teacher and student network output together (see Eq. (3)), the student network could show discrimination towards previously privileged groups, such as males. To alleviate this problem, we update Eq. (3) by adding a parameter $\alpha$ (smaller than 1) to adjust the impact of the teacher network:

$$p(y|x) = \text{softmax}(log(p_S(y|x)) + \alpha log(p_T(y|x))). \quad (5)$$

With $p(y_i|x_i)$ in Eq.(5), the ensemble learning loss is implemented via cross entropy and is given as follows:

$$\mathcal{L}_{\text{IM}}(x) = -\frac{1}{N}\sum_{i=1}^{N} y_i log(p(y_i|x_i)) + (1-y_i)log(1-p(y_i|x_i)). \quad (6)$$

## Overall Loss Function

Putting the above-mentioned two manners of counter-teaching together, i.e., the explicit one in Eq.(1) and the implicit one in Eq.(6), the overall loss function is:

$$\mathcal{L}(x) = \mathcal{L}_{\text{CE}}(x) + \beta_1\mathcal{L}_{\text{EX}}(x) + \beta_2\mathcal{L}_{\text{IM}}(x), \quad (7)$$

where the first term is the standard cross entropy (CE) loss for debiased student prediction $p_S$:

$$\mathcal{L}_{\text{CE}}(x) = -\frac{1}{N}\sum_{i=1}^{N} y_i log(p_S(y_i|x_i)) + (1-y_i)log(1-p_S(y_i|x_i)). \quad (8)$$

The second and third terms are the explicit and implicit decorrelation, respectively. Both are used to suppress the student's reliance on sensitive features. Hyperparameters $\beta_1$ and $\beta_2$ are used to balance these three terms, in order to control the fairness and utility trade-off. Larger $\beta_1$ and $\beta_2$ could lead to reduced discrimination, at the expense of a larger model accuracy drop.

The overall DeFI framework is implemented in two stages. In the first stage, we train the bias-only teacher network $f_T(x)$, and fix its parameters. In the second stage, we use Eq. (7) to train the debiased student network $f_S(x)$. Note that during the second stage, the entire teacher network $f_T(x)$ is fixed, and only the parameters of the student $f_S(x)$ are updated using back-propagation. Ultimately, the teacher network $f_T(x)$ is discarded and only the debiased student network $f_S(x)$ is employed for prediction.

# Experiments

In this section, we conduct experiments to evaluate the effectiveness of the proposed DeFI framework.

| | Adult | MEPS | COMPAS | EEC |
|---|---|---|---|---|
| # **Training instances** | 31600 | 11080 | 3700 | 2940 |
| # **Validation instances** | 4520 | 1482 | 523 | 420 |
| # **Test instances** | 9102 | 3168 | 1055 | 840 |
| **Protected attribute** | Gender | Race | Race | Gender |

Table 1: Dataset Statistics

## Experimental Setup

**Benchmark Datasets.** We use three tabular datasets and one synthetic text dataset. The statistics are given in Tab. 1. The first one is *Adult Income* (Adult), which aims to predict whether a salary is greater than or less than 50K (Kohavi 1996). The second one is *Medical Expenditure* (MEPS). MEPS is a medical dataset aiming to predict whether a person would have high utilization (Bellamy et al. 2018). The third is *COMPAS*, which aims to predict criminal defendant's likelihood of reoffending (Angwin et al. 2016). The fourth one is *Equity Evaluation Corpus* (EEC), which is used to predict the sentiment of texts (Kiritchenko and Mohammad 2018). We differentiate between angry and joy and formulate it into a binary classification problem. To simulate real-world datasets that show discrimination towards certain demographic groups, we manually inject noise into the training dataset to make it biased towards females.

**Details about Protected Attributes.** As shown in Table 1, gender is the protected attribute for the Adult and EEC datasets. The binary attributes include male and female, where female is the unprivileged group for both datasets. Race is selected as the protected attribute for both MEPS and COMPAS datasets. More specifically, binary protected attributes for COMPAS include Caucasian (i.e., European Americans) and African Americans, and African Americans is the unprivileged group. The binary protected attribute for MEPS include Caucasian and Non-Caucasian, where Non-Caucasian is the unprivileged group.

**Baseline Methods.** We compare DeFI with the Vanilla baseline that is trained only by cross entropy loss, as well as the following five baselines methods:

- **Optimized pre-processing (OptimPre)** (Calmon et al. 2017) It is a pre-processing transformation technique to debias the training dataset. The transformation is formulated in a probabilistic framework, where features and labels are edited to ensure group fairness.
- **Adversarial learning (AdverLearn)** (Zhang, Lemoine, and Mitchell 2018) The output layer of the main predictor is used as input to another adversary network. The goal of the predictor is to learn a representation which is maximally informative for the major prediction task, while the role of adversarial classifier is to minimize the predictor's ability to predict the protected attribute.
- **Penalize Explanation (Explanation)** (Liu and Avci 2019) It enforces DNN models to pay more attention to the correct features relevant to the prediction task. The model training is regularized with local DNN interpretation by incorporating annotations from domain experts.
- **Demographic Parity (DP-Gap)** (Bechavod and Ligett 2017) It is implemented as a regularizer, which directly optimizes the metric difference of demographic parity between two protected groups. A hyperparameter is used to control the fairness-accuracy trade off.
- **Equalized Odds Post-processing (EOP)** (Hardt et al. 2016) This is a model-agnostic post-processing method for fairness mitigation. The key idea is to enforce both demographic groups to have the same false positive rate and the same false negative rate.

**DNN Architectures.** Since the focus of this work is on fairness mitigation rather than improving prediction accuracy, we only use standard architectures. We use multilayer perceptron (MLP) for the three tabular datasets (i.e., Adult, MEPS, and COMPAS), and convolutional neural network (CNN) (Kim 2014) for the text dataset (i.e., EEC). The details for the two DNN architectures are given as follows:

- **CNN.** This is a two-dimensional CNN. We perform the convolution operation on the embedding matrix and use convolution of three kernel sizes: $[2, 3, 4]$. After the convolution, we use ReLU activation and max pooling. The resulting tensors will be concatenated as the final representation, which is then connected to the fully connected layer and softmax layer to get the probability output.
- **MLP.** It contains four layers where the node numbers for intermediate layers are 50. We use ReLU after each fully connected layer. Dropout is inserted after the output of the ReLU activation, with a dropout probability of 0.2.

**Implementation Details.** For EEC dataset, we use the 300-dimensional word2vec word embedding (Mikolov et al. 2013) to initialize the embedding layer of the CNN model. The hyperparameter $m$ for Integrated Gradient in Eq.(2) is fixed as 50 for all experiments. The influence weight $\alpha$ in Eq.(5) is set as 0.01, 0.06, 0.03, 0.001 for Adult, MEPS, COMPAS, ECC, respectively. To train the DNN models, we use the Adam optimizer, and the learning rate is searched from {5e-5, 1e-4, 5e-4, 1e-3, 5e-3}. Note that hyper-parameters ($\beta_1$, $\beta_2$) and other hyper-parameters are tuned based on the trade-off between accuracy and fairness metrics on the validation sets.

## Fairness and Accuracy Evaluation

We report fairness-accuracy curves by varying two major hyperparameters of DeFI, i.e., $\beta_1$ and $\beta_2$ in Eq.(7), and varying the degree of regularization for the three in-processing baselines. For the pre-processing and post-processing baselines, we select the best hyperparameters reported in the original paper or official implementations, and report a single point. Random initialization can lead to variance in DNN performance, and thus we report the average values over three runs for all DNNs. The results are given in Fig. 2.

**Comparison with Original DNN.** For the vanilla model trained with only cross entropy loss, i.e., Vanilla, the $\mathcal{F}_{parity}$ values are less than 0.9 for all four datasets. The $\mathcal{F}_{odds}$ differences between two protected groups range from -0.088 to -0.445, implying a discrimination towards the unprivileged groups. For all four datasets, DeFI has consistently improved

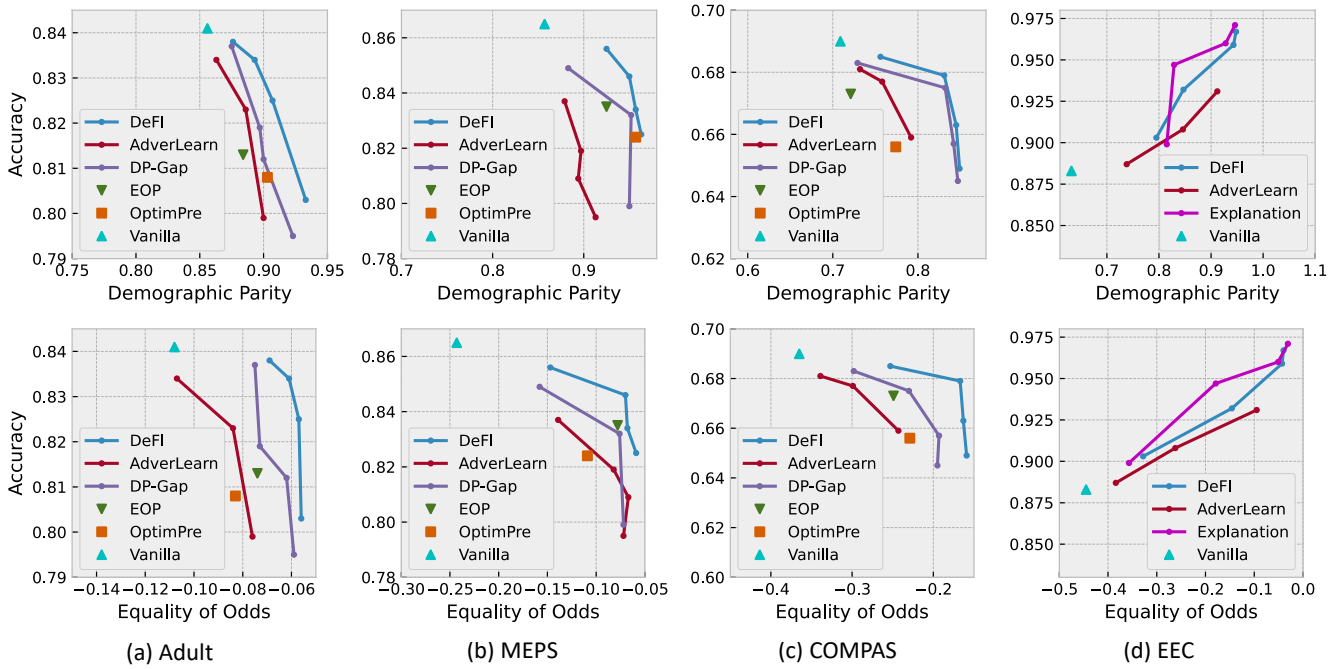



Figure 2: Fairness-accuracy trade off curves. The first row indicates *demographic parity* $\mathcal{F}_{parity}$ metric and the second row denotes *Equality of odds* $\mathcal{F}_{odds}$ metric. Among four datasets, the proposed DeFI achieves consistent performance improvements for two group fairness metrics, while not resulting in large fairness accuracy trade-off. (Best viewed in color)

two fairness metrics. For three tabular datasets, as we increase the values of $\beta_1$ and $\beta_2$, better $\mathcal{F}_{parity}$ and $\mathcal{F}_{odds}$ can be observed, at the expense of a higher accuracy drop.

**Comparison with Other Mitigation Methods.** There are four key findings. First, among four datasets, the proposed DeFI achieves consistent performance improvements for the two group fairness metrics, while not resulting in a large fairness accuracy trade-off. Second, for *pre-processing mitigation*: OptimPre has fewer improvement in terms of two fairness metrics, indicating the limited ability of pre-processing for bias mitigation. Third, in terms of *in-processing mitigation*: AdverLearn can simultaneously improve three fairness metrics. It, however, has come at the expense of relatively lower accuracy. When it has similar accuracy to DeFI, it has larger discrimination. One possible explanation is that adversarial learning could potentially remove other useful information that the model could rely on to make predictions. The Explanation baseline could achieve comparable performance on EEC with DeFI. However, this method requires annotating an exhaustive list of sensitive features, which is impractical in many applications. Fourth, for *post-processing mitigation*: EOP has consistent improvement for both fairness metrics. Nevertheless, EOP possesses two limitations: 1) dramatic accuracy drop and 2) requiring the testing time access to protected attributes. This is usually not practical in real-world applications to get access to protected attributes, thus reducing the applicability of post-processing bias mitigation methods.

**Why No Fairness-Accuracy Tradeoff for EEC Dataset?** In Fig. 2, DeFI has sacrificed accuracy for Adult, MEPS, and COMPAS datasets, while improveing accuracy for EEC dataset. The main reason is that the distributions for the training and test set are the same for Adult, MEPS, and COMPAS, where the fairness sensitive features are predictive of labels both on the training and test sets. In contrast, for EEC, we only inject noise into the training set. As a result, those fairness sensitive features are only predictive of labels in training set and have no connection with labels on the test set. The improved accuracy of EEC also validates that DeFI has successfully decorrelated the connection between fairness sensitive features with main task labels.

| | Accuracy | Race Bias | | | Gender Bias | | |
|---|---|---|---|---|---|---|---|
| **Models** | $\mathcal{F}_{acc}$ | $\mathcal{F}_{parity}$ | $\mathcal{F}_{opty}$ | $\mathcal{F}_{odds}$ | $\mathcal{F}_{parity}$ | $\mathcal{F}_{opty}$ | $\mathcal{F}_{odds}$ |
| Vanilla | 86.5 | 0.857 | -0.195 | -0.243 | 0.938 | -0.052 | -0.076 |
| DeFI | 84.6 | 0.950 | -0.057 | -0.070 | 0.961 | -0.081 | -0.095 |
| DeFI_comb | 84.2 | 0.955 | -0.061 | -0.076 | 0.983 | -0.032 | -0.036 |

Table 2: Compositional fairness.

## Compositional Fairness

We use MEPS dataset to investigate the mitigation of compositional fairness (combination of multiple sensitive attributes (Bose and Hamilton 2019)), since it has available labels for two attributes: gender and race. We fix hyperparameters ($\beta_1$, $\beta_2$) as (1.5, 3), and report a single point on fairness accuracy curve as in Tab. 2.

**Limitation of Regularizing One Attribute.** In real-world applications, there usually exists more than one protected attribute. The reduction in bias of one attribute could increase the bias of another attribute. Take MEPS dataset for

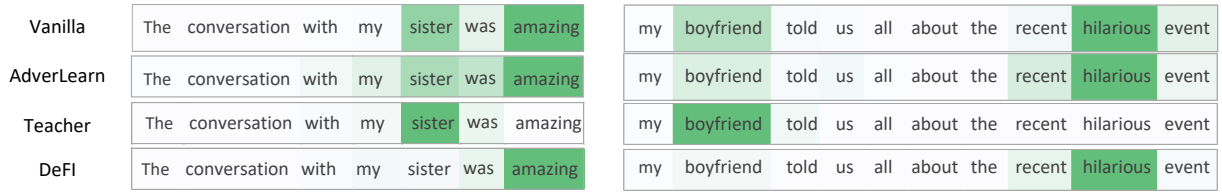



Figure 3: Two illustrative examples of interpretations. The proposed method DeFI could mainly focus on task-relevant sentiment features, i.e., '*amazing*' and '*hilarious*', for prediction.

| | Vanilla | AdverLearn | Teacher | DeFI |
|---|---|---|---|---|
| $\mathcal{F}_{bias}$ | 0.35 | 0.21 | 2918.52 | 0.05 |

Table 3: Interpretation ratio

example, as shown in Tab. 2. The regularization of the race attribute has improved model (i.e., DeFI) performance for the race attribute. However, DeFI at the same time sacrifices some fairness metrics for the gender attribute ($\mathcal{F}_{opty}$ from -0.052 to -0.081 and $\mathcal{F}_{odds}$ from -0.076 to -0.095). This is because DNN models tend to take *shortcuts* to make predictions. The reduced attention of one shortcut (i.e., race) might amplify model's reliance on other shortcuts (e.g., race).

**Compositional Fairness.** We extend DeFI to compositional fairness by training multiple biased teacher models for several protected attributes. For MEPS dataset, we train two biased teachers for race and gender attributes. As shown in Tab. 2, the DeFI_comb model has improved three fairness metrics for both race and gender attributes compared to Vanilla model. More encouragingly, there is only a 0.4% accuracy drop for DeFI_comb compared to DeFI. This conforms that DeFI can mitigate discrimination towards multiple protected attributes, with a negligible drop in accuracy.

## Interpretation for Sanity Check

We quantitatively and qualitatively analyze the connections of interpretation with model bias.

**Visualizations for EEC Dataset.** We visualize interpretations for 4 comparing methods in Fig. 3. There are three key findings. First, the teacher network highlights all fairness sensitive features, such as '*sister*' and '*boyfriend*'. This is a major advantage of the teacher network, where it could tell us not only which subsets of features are highly relevant to protected attributes, but also the corresponding likelihood. Second, the Vanilla model focuses comparable attention on fairness sensitive features and task-relevant features. Third, the debiased DeFI learns to pay less attention to fairness sensitive features. Instead, DeFI mainly captures more task-relevant features for prediction, i.e., '*amazing*' and '*hilarious*'. This demonstrates DeFI has captured complementary information as the teacher network.

**Quantitative Evaluation for EEC.** We manually select out fairness sensitive features (e.g., *'she', 'sister', 'he', 'brother'*) and task-relevant features (e.g., *'excited', 'wonderful', 'angry', 'annoyed'*) from EEC dataset. Then the bias degree of the models is defined as the average ratio between the importance values of interpretation of two list of features: $\mathcal{F}_{bias} = \frac{1}{n}\sum_{i=1}^{n}\frac{p_{sensitive}}{p_{task}}$, where the smaller $\mathcal{F}_{bias}$ is, the less attention is paid from the model to fairness sensitive features. The results are reported in Tab. 3. It indicates that original DNN pays comparable attention, i.e., 0.35, to fairness sensitive features and task-relevant features. This results in the over-association between demographic with certain labels, leading to its discrimination behavior. The teacher network mainly focuses on sensitive features, with a ratio of 2918.52. Benefiting from this teacher network, DeFI substantially reduces the attention of the student network for fairness sensitive features (from 0.35 to 0.05).

| | Accuracy | Race Bias | | |
|---|---|---|---|---|
| **Models** | $\mathcal{F}_{acc}$ | $\mathcal{F}_{parity}$ | $\mathcal{F}_{opty}$ | $\mathcal{F}_{odds}$ |
| Vanilla | 86.5 | 0.857 | -0.195 | -0.243 |
| DeFI | 84.6 | 0.950 | -0.057 | -0.070 |
| DeFI_explicit | 84.3 | 0.956 | -0.061 | -0.078 |
| DeFI_implicit | 86.0 | 0.938 | -0.037 | -0.063 |

Table 4: Ablation analysis.

## Ablation Analysis

DeFI has two components for counter-teaching from the teacher network: DeFI_explicit and DeFI_implicit. We use MEPS dataset to conduct ablation studies to analyze their contributions, and report the results in Tab. 4 (we fix hyper-parameters ($\beta_1$, $\beta_2$) as (1.5, 3)). There are two main findings. Firstly, both DeFI_explicit and DeFI_implicit could improve the model with regard to all three fairness metrics. Secondly, DeFI_explicit and DeFI_implicit bring different benefits. DeFI_explicit has more improvement for the demographic parity $\mathcal{F}_{parity}$, while DeFI_implicit has more improvement for both $\mathcal{F}_{opty}$ and $\mathcal{F}_{odds}$. Besides, DeFI_implicit has relatively higher accuracy than DeFI_explicit.

## Conclusions

In this work, we propose a bias mitigation framework, called DeFI, to decorrelate influence of fairness sensitive features for the prediction task. DeFI first trains a bias-only teacher network and then counter-teaches a debiased student network to encourage the student to downweight its attention to sensitive features. DeFI is model-agnostic, easy to implement, and does not require access to protected attributes at test time. Despite the simplicity, we show that DeFI could increase the DNN performance with respect to three group fairness measurements, with a negligible drop in accuracy.

## Acknowledgments

The work is in part supported by NSF grants CNS-1816497, IIS-1900990, and IIS-1939716. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing any funding agencies.

## References

Angwin, J.; Larson, J.; Mattu, S.; and Kirchner, L. 2016. How We Analyzed the COMPAS Recidivism Algorithm. In *ProPublica*.

Bechavod, Y.; and Ligett, K. 2017. Penalizing unfairness in binary classification. *4th Workshop on Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*.

Bellamy, R. K.; Dey, K.; Hind, M.; Hoffman, S. C.; Houde, S.; Kannan, K.; Lohia, P.; Martino, J.; Mehta, S.; Mojsilovic, A.; et al. 2018. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. *arXiv preprint arXiv:1810.01943*.

Bose, A. J.; and Hamilton, W. 2019. Compositional Fairness Constraints for Graph Embeddings. *International Conference on Machine Learning (ICML)*.

Calmon, F.; Wei, D.; Vinzamuri, B.; Ramamurthy, K. N.; and Varshney, K. R. 2017. Optimized pre-processing for discrimination prevention. In *Conference on Neural Information Processing Systems (NIPS)*.

Clark, C.; Yatskar, M.; et al. 2019. Don't Take the Easy Way Out: Ensemble Based Methods for Avoiding Known Dataset Biases. *The 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Du, M.; Liu, N.; and Hu, X. 2020. Techniques for interpretable machine learning. *Communications of the ACM (CACM)*.

Du, M.; Yang, F.; Zou, N.; and Hu, X. 2020. Fairness in Deep Learning: A Computational Perspective. *IEEE Intelligent Systems*.

Feldman, M.; Friedler, S. A.; Moeller, J.; Scheidegger, C.; and Venkatasubramanian, S. 2015. Certifying and removing disparate impact. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*.

Gajane, P.; and Pechenizkiy, M. 2018. On formalizing fairness in prediction with machine learning. *5th Workshop on Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*.

Geirhos, R.; Jacobsen, J.-H.; Michaelis, C.; Zemel, R.; Brendel, W.; Bethge, M.; and Wichmann, F. A. 2021. Shortcut Learning in Deep Neural Networks. *Nature Machine Intelligence*.

Hardt, M.; Price, E.; Srebro, N.; et al. 2016. Equality of opportunity in supervised learning. In *Conference on Neural Information Processing Systems (NIPS)*.

He, H.; Zha, S.; and Wang, H. 2019. Unlearn dataset bias in natural language inference by fitting the residual. *2019 EMNLP workshop*.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *NIPS 2014 Deep Learning Workshop*.

Hinton, G. E. 2002. Training products of experts by minimizing contrastive divergence. *Neural computation*.

Kim, Y. 2014. Convolutional neural networks for sentence classification. *2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Kiritchenko, S.; and Mohammad, S. M. 2018. Examining gender and race bias in two hundred sentiment analysis systems. *Proceedings of the 7th Joint Conference on Lexical and Computational Semantics*.

Kohavi, R. 1996. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*.

Liu, F.; and Avci, B. 2019. Incorporating Priors with Feature Attribution on Text Classification. *57th Annual Meeting of the Association for Computational Linguistics (ACL)*.

McDonnell, T.; Lease, M.; Kutlu, M.; and Elsayed, T. 2016. Why is that relevant? Collecting annotator rationales for relevance judgments. In *Fourth AAAI Conference on Human Computation and Crowdsourcing*.

Mehrabi, N.; Morstatter, F.; Saxena, N.; Lerman, K.; and Galstyan, A. 2022. A survey on bias and fairness in machine learning. *ACM Computing Surveys*.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; et al. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*.

Nagpal, S.; Singh, M.; Singh, R.; Vatsa, M.; and Ratha, N. 2019. Deep Learning for Face Recognition: Pride or Prejudiced? *arXiv preprint arXiv:1904.01219*.

Phuong, M.; and Lampert, C. 2019. Towards understanding knowledge distillation. In *International Conference on Machine Learning (ICML)*.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic Attribution for Deep Networks. *International Conference on Machine Learning (ICML)*.

Tang, R.; Du, M.; Li, Y.; Liu, Z.; Zou, N.; and Hu, X. 2021. Mitigating Gender Bias in Captioning Systems. In *Proceedings of the Web Conference (WWW)*.

Wan, M.; Zha, D.; Liu, N.; and Zou, N. 2021. Modeling Techniques for Machine Learning Fairness: A Survey. *arXiv preprint arXiv:2111.03015*.

Wu, F.; Du, M.; Fan, C.; Tang, R.; Yang, Y.; Mostafavi, A.; and Hu, X. 2021. Understanding Social Biases Behind Location Names in Contextual Word Embedding Models. *IEEE Transactions on Computational Social Systems*.

Zafar, M. B.; Valera, I.; Gomez Rodriguez, M.; and Gummadi, K. P. 2017. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the Web Conference (WWW)*.

Zhang, B. H.; Lemoine, B.; and Mitchell, M. 2018. Mitigating unwanted biases with adversarial learning. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (AIES)*.