Algorithmic Fairness in Machine Learning

Mengnan Du, Lu Cheng, Dejing Dou

Abstract As machine learning models are increasingly being deployed in real-world applications, these models would exhibit bias toward specific demographic groups. In this chapter, we discuss the topic of algorithmic fairness in machine learning. First, we divide the existing fairness notation into two broad groups and introduce some representative notation. Then, we introduce three categories of fairness mitigation algorithms, i.e., pre-processing, in-processing, and post-processing mitigation methods. Finally, we discuss research challenges, potential future research directions, and the relationship between fairness in machine learning and other areas.

1 Definitions of Fairness

Nowadays, machine learning models have made remarkable breakthroughs in a number of fields, due to developments in complex models such as deep neural networks (DNNs) and the collection of numerous large-scale datasets. Machine learning models are increasingly being used in real-world applications that interact with end users, such as healthcare, recommender system, criminal justice, recruitment, etc [53]. Recent studies indicate that these machine learning models might exhibit discrimination behavior for certain demographics. For example, the error rates of darker-skinned females (up to 34.7%) are much higher than those of lighter-skinned males (up to 34.7%) [5]. This dramatic accuracy disparity could cause significant damage to the group of darker-skinned women. Similarly, the AI-based recruiting tool of a tech company shows discrimination against women by penalizing resumes with the keyword 'women'. The bias and unfair behavior of real-world AI systems could cause significant harm to individuals and our society. As such, there is growing interest from both academia and industry in addressing the problem of unfairness [59, 63, 19].

1.1 What Does It Mean by Fairness?

It is not trivial to give the definition of fairness, since fairness is a broad topic and its definitions are significantly dependent on specific application domains. We first analyze two case studies below.

Case Study 1: Predicting the length of stay. Suppose that a hospital wants to develop an AI system to predict the length of stay for inpatients. More resources should be given to hospitalized inpatients who are expected to be discharged sooner to avoid delays. The system might predict that residents of ZIP codes in African-American neighborhoods will stay longer. As a result, the hospital would allocate more resources to European Americans and much fewer resources to African Americans [48].

Case Study 2: Voice recognition of ASR systems. Automated speech recognition (ASR) algorithms are used to translate spoken language into text. They have been applied in many commercial products, such as virtual assistants of smart speakers, automated closed captioning, etc. There are noticeable racial disparities in these ASR systems, according to recent studies [34]. In particular, African-American speakers have a word error rate on average of 0.35, which is significantly higher than the word error rate on average for European Americans, which is 0.19.

We can see from these two case studies that a system could have a different impact on humans of different demographics, including gender, race, sexual orientation, and age, to name a few. From the perspective of impact, we can group the harms that groups or individuals can receive into the following two groups [15, 3].

- **Resources and Opportunities Allocations Harm:** This corresponds to Case Study 1. These harms can occur in high-stake applications that involve opportunities, resources, or information. Representative applications include employment, criminal justice, education, etc. The impact often indicates whether end users will have access to resources and opportunities, such as receiving a job offer from a company, having a house loan granted, or getting accepted into a school. Some underprivileged groups, such as women and African Americans, would suffer more allocation harm in this situation.
- **Quality of Services Harm:** This corresponds to Case Study 2. This kind of harm is more prevalent in some service settings, such as speech recognition, facial recognition, machine translation, etc. The bias problem corresponds to *service quality* that an end user can receive. Machine learning techniques do not always benefit all demographic groups equally, and certain underprivileged populations have substantially lower prediction accuracy. For instance, African Americans have substantially worse prediction accuracy than European Americans for applications such as speech recognition and facial recognition.

Table 1 Categorization of the fairness problem in machine learning and representative examples, where the examples are taken from [15]. We classify fairness problems into two categories: prediction outcome discrimination and prediction quality disparity. The first category would make resource allocations and opportunities harmful. In contrast, the second category would cause harm to the quality of services.

Class	Representative examples
Outcome Discrimination	*Employment*: The recruiting tool believes that men are more qualified and shows bias against women. *Loan Approval*: The loan eligibility system negatively rates people belonging to certain ZIP code, causing discrimination for certain races. *Criminal Justice*: The recidivism prediction system predicts that black inmates are three times more likely to be classified as 'high risk' than white inmates.
Quality Disparity	*Facial Recognition*: Facial recognition performs very poorly for female with darker skin. *Language processing*: Language identification models perform significantly worse when processing text produced by people belonging to certain races.

1.2 Two Families of Algorithmic Fairness

Based on the harms that people could receive, we group fairness measurements into two broad categories from the machine learning model prediction perspective. It includes (i) prediction outcome discrimination and (ii) prediction quality disparity, which correspond to the two types of harm introduced in the last section. More representative examples are given in Table 1.

Before covering the fairness measurements, we first introduce the notation used in this chapter. We consider the typical classification problem using labeled examples: $\{x, y, a\} \sim p_{data}$. Here, $x \in \mathcal{X}$ denotes the input feature and $y \in \mathcal{Y}$ represents the label that we want to predict. Furthermore, $a \in \mathcal{A} = \{0, ..., K\}$ is a K categorical *protected attribute* annotation, such as race, gender, and age. For protected attributes, we assume that there exist certain unprivileged groups and privileged groups where we denote $a = 0$ and $a = 1$ as the unprivileged group and the privileged group, respectively. Take a loan application as an illustration. The unprivileged group $a = 0$ could be African Americans and the privileged group $a = 1$ would include European Americans. The privileged group denotes a group that has historically enjoyed a systematic advantage. Here, the goal is to learn the classification model that can be denoted as $f(x) : \mathcal{X} \to \mathcal{Y}$. The model $f(x)$ can be any of the machine learning models, e.g., the traditional machine learning model such as random forests or deep neural network models (DNNs) such as convolutional neural networks.

Prediction Outcome Discrimination. This is the most widely used group fairness family that expects parity in statistical performance between groups. It tries to minimize the gap between groups and aims to optimize fairness metrics such as demographic parity and equality of opportunity. We introduce the three most widely used parity-based group fairness metrics in this category: demographic parity, equality of opportunity, and equality of odds.

Demographic parity [18] calculates the favorable outcome gap between the unprivileged group ($a = 0$) and the privileged group ($a = 1$):

$$\mathcal{F}_{DP} = p(\hat{y} = 1|a = 0) - p(\hat{y} = 1|a = 1), \tag{1}$$

where $\hat{y}$ is the model prediction and 1 denotes the favorable outcome, such as obtaining a loan, receiving an offer of work, and avoiding jail. In addition to calculating the gap, some other works formulate it into the ratio between two groups.

$$\mathcal{F}_{DP} = \frac{p(\hat{y} = 1|a = 0)}{p(\hat{y} = 1|a = 1)} \tag{2}$$

Equality of opportunity metric [28, 65] is defined as the true positive rate difference between the unprivileged group and the privileged group:

$$\mathcal{F}_{EOP} = p(\hat{y} = 1|a = 0, y = 1) - p(\hat{y} = 1|a = 1, y = 1). \tag{3}$$

Unlike demographic parity, the calculation of the equality of opportunity depends on the ground truth label y.

Equality of odds metric [28] also takes into account the false positive rate:

$$\mathcal{F}_{EOO} = p(\hat{y} = 1|a = 0, y = 0) - p(\hat{y} = 1|a = 1, y = 0) + \mathcal{F}_{EOP}. \tag{4}$$

For fair models, the value of all three parity-based metrics should be as close to 0 as possible if we calculate the gap between two groups. A larger gap denotes greater discrimination for the unprivileged group. Similarly, the closer the ratio between two groups is to one, the better the model.

The three aforementioned measurements are defined from the perspective of group fairness. On the other hand, we can also require the fairness to be satisfied from the *individual fairness* perspective. This requires that individuals with similar profiles (especially those with different demographic groups) are treated similarly [23]. The most common definition is as follows:

$$\text{iff } p(x_i) \approx p(x_j)||d(x_i, x_j) \approx 0, \tag{5}$$

where x_i and x_j denote individuals, and $d : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a distance metric to quantify the similarity of individuals. Note that the distance metric should be carefully designed to fit the characteristics of the underlying task [16].

Prediction Quality Disparity. It is another widely used group fairness family, which can also be called Rawlsian Max-Min fairness. A typical example is that darker skin-tone female groups have much higher misclassification rates than lighter skin-tone male groups. Similarly, the error rate for black speakers is much higher than that for white speakers for automated speech recognition (ASR) systems. The goal of Max-Min fairness is to improve the worst group performance [37].

$$\max_{a \in \mathcal{A}} \mathcal{U}(\hat{y}, y, a = 0). \tag{6}$$

Here $\mathcal{U}$ is the target utility metric for the task, such as accuracy, precision, recall, F1, etc. The goal is to maximize the performance $\mathcal{U}$ of the group with the poorest performance $a = 0$ (such as women) without sacrificing the performance of other groups (such as men). Alternatively, we can optimize the performance difference between the privileged group and the unprivileged group.

$$\mathcal{F}_{QP} = \mathcal{U}(\hat{y}, y, a = 0) - \mathcal{U}(\hat{y}, y, a = 1). \tag{7}$$

Here, the absolute value of $\mathcal{F}_{QP}$ is supposed to be as small as possible for fair models. The goal of both formulations is to improve the quality of the prediction of the unprivileged group, thus alleviating the quality of the service harm to them.

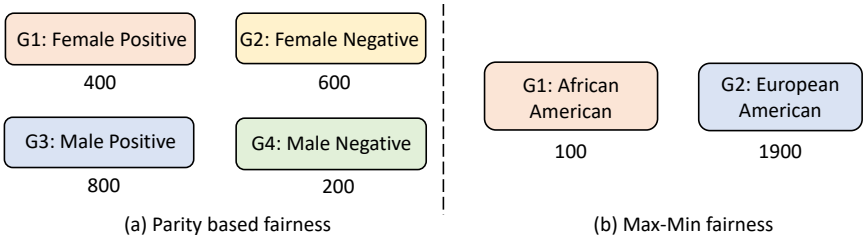

Fig. 1 Illustration of difference between (a) prediction outcome discrimination and (b) prediction quality disparity using toy examples. Suppose that we have 2,000 training samples in total, and the number below each box indicates the training number for that category. For prediction outcome discrimination (a), we can group the existing training data into 4 groups: females with positive label, females with negative label, males with positive label and males with negative label. The training set has imbalanced distribution, where females are more associated with negative label, while males are more associated with positive label. A desirable model is to rely on task-relevant features for prediction. In this case, however, the trained models would over-associate the fairness sensitive information relevant to females with negative labels, and vice versa. It can be explained by the simplicity bias, where the fairness sensitive features are simple and highly correlated with class labels [52]. Thus the model will highly rely on them for prediction, which would result in the discrimination of the model towards females and lead to the statistical parity difference between two groups. In contrast, for (b) prediction quality disparity, the unprivileged group, e.g., African Americans here, has much fewer training samples compared to the privileged group, e.g., European Americans. This is similar to the long-tailed classification scenario, where the unprivileged group corresponds to the tail category. Machine learning models are designed to optimize for the overall performance. If they can not simultaneously optimize for all groups, they will optimize for the majority groups instead. As a result, the models will have poor prediction performance for the unprivileged group.

1.3 Origins of Bias

In this section, we discuss the origins of algorithmic unfairness. Bias of many different kinds can result in unfair algorithms. Be aware that bias in machine learning is a broader topic. The texture bias in the convolutional neural network (CNN) is a

Table 2 The origins of algorithmic bias. Here, we introduce the three most representative types of bias. The sampling bias is from the data perspective, while the other two types of bias are from the modeling perspective.

	Sampling Bias	Amplification Bias	Underrepresentation Bias
Outcome Discrimination	✓	✓	
Quality Disparity	✓		✓

typical example [26, 29]. Compared to bias in machine learning, fairness is a more specialized field because only biases that are important to humans may be referred to as fairness problems. We only discuss the three bias families that are most likely to lead to the fairness problem (see Table 2), which can be used to explain most types of algorithmic unfairness, as discussed in the previous section.

- **Sampling Bias.** This is also called *selection bias* in the literature [44]. This is from the data perspective. Sampling bias occurs when the data set's examples are chosen in a way that is not representative of the real-world distribution of the data. Sampling bias can appear in a variety of ways. Both prediction outcome discrimination and quality disparity can be explained from the perspective of selection bias. However, there are some significant differences, and we illustrate the difference using the toy examples in Figure 1. More specifically, for the discrimination of the prediction outcome, the proportion of women with positive labels is much lower compared to that of men. As a result, models tend to learn that the unprivileged woman group is correlated with a negative label and vice versa. In contrast, for the prediction quality disparity, the number of African American training samples is much smaller than that of European Americans.
- **Algorithmic Amplification Bias.** It denotes the tendency of machine learning models to amplify the biases present in the data on which they are trained [51]. This applies mainly to discrimination by prediction outcome. Unlike certain other types of bias, it is the result of the algorithm and cannot be exclusively attributed to the bias of the dataset [61]. For example, the likelihood of cooking images that contain females is twice that of those that contain males in the training set. After model training, machine learning models amplify this disparity five times [67].
- **Underrepresentation Bias.** In real-world applications, data for some segments of the population can be collected less informatively or more imperfectly. Furthermore, due to the difference in characteristics between different protected groups, machine learning models sometimes cannot optimize all groups simultaneously [15]. As a result, the model would optimize for the majority group (i.e., the privileged group). This results in a poor representation captured by the model for the minority group (i.e., the unprivileged group). Eventually, the models would have a lower prediction quality for the underrepresented group.

Prediction Outcome Discrimination v.s. Quality Disparity Based on the analysis in previous subsections, we summarize the significant differences between the two families of fairness notation.

First, the causing reasons are different. For prediction outcome discrimination, due to sampling bias, the privileged group is more correlated with the favorable label and vice versa. Machine learning models have captured this kind of data bias and even amplify data bias through spurious learning (i.e., shortcut learning) [25]. Specifically, there are usually both fairness-sensitive features and task-relevant features in the input. The models have over-associated these fairness-sensitive features with certain class labels, rather than learning the underlying task through task-relevant features. As a result, the model would show discrimination against the unprivileged groups. The prediction quality disparity is typically due to the long-tailed classification problem. Specifically, the tail of the distribution contains data from the unprivileged group and has only very few training samples. Consider the English-based speech recognition problem as an example. There are much fewer training data from non-White groups (such as Asians) in the training corpus than from White groups. Therefore, the models are optimized for the privileged group and perform poorly for the underprivileged groups.

Second, although both groups of fairness concepts mean that the profit of the unprivileged groups has been sacrificed, the application domains are different, and thus the impacts are different (see Section 1.1). For the discrimination of the prediction outcomes, there generally exist two opposite labels y, where $y = 1$ denotes the desirable outcome and $y = 0$ denotes the undesirable outcome. Unprivileged groups tend to get predictions of an undesirable outcome. Thus the impact for them is the withholding of resources and opportunities. In contrast, there does not necessarily exist an opposite outcome $y = 0$ and $y = 1$ for the prediction quality disparity problem. In addition, it typically affects the quality of service and thus hurts the user experience of unprivileged groups.

2 Bias Mitigation Algorithms

In this section, we present mitigation algorithms that aim to alleviate the bias issue of machine learning models and improve fairness. A typical machine learning life cycle contains three stages: data collection and preparation, model training, and model deployment. As such, we group fairness mitigation methods into three categories based on the machine learning life cycle, including pre-processing, in-processing, and post-processing mitigation approaches [2].

We further divide mitigation methods into prediction outcome discrimination and prediction quality disparity, due to their significant differences in the mechanisms. Furthermore, it is assumed that existing mitigation methods would result in a fairness and utility trade-off for discrimination of the prediction outcome. On the contrary, prediction quality disparity mitigation algorithms could simultaneously improve fairness and utility.

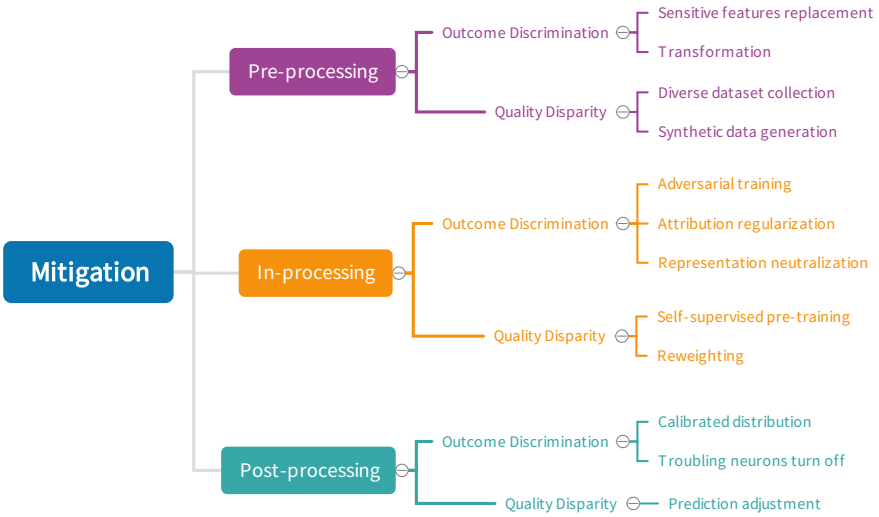

Fig. 2 Mitigation methods and corresponding examples. Based on the typical machine learning life cycle, we group the mitigation methods into pre-processing, in-processing, and post-processing three categories. Based on this, we further split methods into Outcome Discrimination Quality Disparity two groups. For each sub-group, we list several representative examples.

2.1 Pre-processing Methods

Both problems of discrimination of the prediction outcomes and the disparity in prediction quality are caused by unbalanced and skewed training sets. To address this issue, pre-processing methods are proposed with the goal of creating more high-quality training data.

2.1.1 Prediction Outcome Discrimination

The prediction outcome discrimination is mainly caused by the imbalanced conditional distribution of fairness-sensitive features in the input with class labels. First, one straightforward idea is to remove these fairness features, e.g., ZIP code in tabular dataset applications. Second, some mitigation methods replace fairness-sensitive features with alternative values, as removing features is not possible in many applications. For example, removing words from texts could cause grammar errors. The use of counterfactual data augmentation to build a balanced training set is a natural mitigation technique within this family. It might, to some extent, reduce the degree of discrimination of trained machine learning models. For example, an auxiliary dataset is created by replacing male entities with female entities and vice versa [68]. The auxiliary dataset and the original dataset can be combined to create a balanced dataset. Models trained on this balanced dataset significantly alleviate bias, while

not sacrificing the original task performance. It is also worth noting that counterfactual data augmentation could result in an increased balance in terms of explicit sensitive features. However, discrimination can still be caused by those implicit sensitive features. It is extremely challenging to alleviate all kinds of statistical shortcut cue that the model could exploit. On the other hand, counterfactual data augmentation also requires massive domain expertise to ensure that the augmented data are reasonable and fall into the original training data distribution. As a result, machine learning models could still capture information such as gender, race in intermediate representation and eventually show discrimination for the unprivileged group.

2.1.2 Prediction Quality Disparity

The reason for the quality disparity problem is that the unprivileged group has much fewer training data compared to the privileged group. Therefore, data sampling and data augmentation can be used to make the training set more balanced. First, data sampling can be used to increase the relative ratio of the unprivileged group. As such, we can have better representations for the unprivileged group. However, this has the risk of deteriorating the performance of the privileged group. Second, the data augmentation method can be used to collect more training data for the unprivileged group. During the data collection process, crowd workers are encouraged to collect data from a diverse data source. Consider the racial disparity in speech recognition application; we can use more varied training datasets that include African-American vernacular English [34]. However, this could be costly in practice. Alternatively, generative adversarial networks (GANs) can be used to create synthetic data for underrepresented and unprivileged groups [22].

2.2 In-processing Methods

In this section, we introduce in-processing based mitigation methods. These methods add auxiliary regularization terms during the model training process to the overall objective function, explicitly or implicitly regularizing the model to achieve certain fairness metrics.

2.2.1 Prediction Outcome Discrimination

For the discrimination of prediction outcomes, we focus on introducing mitigation methods that are applicable to deep neural networks (DNNs).

Consider that the classification model $f(x)$ can be represented as $f(x) = c(g(x))$. Here, $g(x) : \mathcal{X} \to \mathcal{Z}$ represents the feature encoder and $g(x) = z$ is the representation of x obtained from a DNN model. The predictor $c(z) : \mathcal{Z} \to \mathcal{Y}$ is the multi-layer classification head. It is represented by the top layer(s) of the DNN, which takes the

encoded representation z as input and maps it to the softmax probability. The final prediction of the model is indicated by $\tilde{y} = \arg\max c(z)$. It is worth noting that the split of the model into encoder and classification head depends on the architecture of the model and the classification task at hand. For example, we can take the 12-layer BERT-base [12] model as the feature encoder and the last two fully connected layers as the classification head. Similarly, for instance, consider the 19-layer VGG model for image classification tasks [54]. We can take the first 16 convolution layers as the encoder and the last 3 fully connected layers as the classification head.

Debiasing Representations. A line of methods attempts to learn debiased representations $g(x)$, which do not contain information about sensitive attributes. The most representative example is adversarial training [58, 62, 17, 66]. In addition to the classification head $c(z)$, we also need an adversarial classifier $h(z)$, which is utilized to predict protected attributes z. The adversarial training process is indicated below.

$$\begin{array}{ll} \arg\min_h & L(h(g(x)),a) \\ \arg\min_{h,c} & L(c(g(x)),y) - \lambda L(h(g(x)),a) \end{array} \tag{8}$$

In the first step, we train the adversarial classifier $h(z)$ by maximizing its ability to predict the protected attribute. In the second step, we train the task classifier $c(z)$ together with the adversarial classifier $h(z)$, where the goal is to maximize the ability of the task classifier to predict the task label y while minimizing the ability of the adversarial classifier $h(z)$ to predict the protected attribute. In this way, sensitive information can be partially removed from the representation $g(x)$. Adversarial training is advantageous in that it is model-agnostic and can be used in any application where DNNs are the classification model, e.g., image classification, text classification, etc. However, this line of methods may not be stable in training. In addition, it could suffer from a large fairness and utility trade-off.

Debiasing Classification Head. Some methods propose to learn the debiased task classification head $c(z)$. The key motivation is that the classification head is much more time-efficient to be debiased, compared to debiasing the representation. For example, a work aims to reduce discrimination of DNN models by only debiasing the classification head $c(z)$, with the biased representation encoder $g(x)$ as input [14]. When training the classification head, for an input sample $\{x_1, y, a_1\}$, they randomly select another sample $\{x_2, y, a_2\}$, with the same class label y but with a different sensitive attribute a_2 compared to a_1 in the input sample. Then they calculate the corresponding representations $z_1 = g(x_1)$ and $z_2 = g(x_2)$ and retrain the classification head using the neutralized representation $z = \frac{z_1+z_2}{2}$ as input. For the supervision label y for the classification head, they use the neutralized soft probability $y = \frac{p_1+p_2}{2}$ after temperature scaling. Given the logit vector z_1 for input x_1, the probability of class i is calculated as $p_{1,i} = \frac{\exp(z_{1,i}/T)}{\sum_j \exp(z_{1,j}/T)}$, where $T \geq 1$. With the neutralized representation and neutralized soft probability, the classification head is trained using the mean squared error (MSE) loss function.

$$\mathcal{L}_{\text{MSE}} = (\hat{y}_i - y)^2 = \left\{ c\left(\frac{1}{2}z_1 + \frac{1}{2}z_2\right) - \left(\frac{1}{2}p_1 + \frac{1}{2}p_2\right) \right\}^2. \tag{9}$$

The aforementioned training program has two key advantages. From the viewpoint of the input, neutralizing the representations prevents the model from capturing the unfavorable association between the information in the representation that is fairness-sensitive and the class labels. From the viewpoint of the output, the softened label encourages the model to make similar predictions for various sensitive groups.

To further enforce the model to ignore sensitive attributes, they construct augmented training samples using a hyperparameter λ to control the degree of neutralization of the samples $\{z_1, p_1, y\}$ and $\{z_2, p_2, y\}$. The augmented neutralized sample is given by $z = \lambda z_1 + (1-\lambda)z_2$, $\lambda \in [\frac{1}{2}, 1)$. They encourage the classification head to give similar prediction scores for the augmented and neutralized sample (with $\lambda = \frac{1}{2}$). The regularization loss is given by:

$$\mathcal{L}_{\text{Smooth}} = \sum_{\lambda \in [\frac{1}{2}, 1)} |c(\lambda z_1 + (1-\lambda)z_2) - c(\frac{1}{2}z_1 + \frac{1}{2}z_2)|_1. \tag{10}$$

Varying λ can control the degree of sensitive information for augmented samples. It is utilized to penalize large changes in softmax probability as we move along the interpolation between two samples. The final loss function is the linear combination of the MSE loss with the regularization term as follows:

$$\mathcal{L} = \mathcal{L}_{\text{MSE}} + \alpha \mathcal{L}_{\text{Smooth}}. \tag{11}$$

The classification head is trained using the loss function in Eq. (11), where α controls the degree of smoothness. Eventually, the original encoder and re-trained classification head are combined as the final debiased network.

Some other methods try to debias the whole classification model. These methods add a regularizer to the model, where the parameters of the entire model are updated. Under this umbrella, we introduce three typical methods: explainability-based regularization, direct regularization of fairness metrics, and implicit regularization.

Debiasing Entire Model: Explainability-based Method. First, one of the most representative methods is to regularize the local explanation of the model $f_{\text{loc}}(x)$ with annotations from the domain expert [39]. The general format of the loss function can be denoted as follows.

$$L(\theta, x, y, r) = \underbrace{d_1(y, \hat{y})}_{\text{Prediction}} + \underbrace{\lambda_1 d_2\left(f_{\text{loc}}(x), r\right)}_{\text{Fairness}} + \underbrace{\lambda_2 \mathcal{R}(\theta)}_{\text{Requlizer}}, \tag{12}$$

where the first and third terms are the standard loss functions to train the DNN models, such as cross-entropy loss. The second term is the additional regularizer to achieve fairness. The two hyperparameters λ_1 and λ_2 are used to balance the three terms. Here, r are the sensitive feature annotations of the domain expert. $f_{\text{loc}}(x)$ is the local explanation of the model for the input sample x that could be implemented using explanation algorithms [13] such as the integrated gradient method [55]. More specifically, $f_{\text{loc}}(x)$ is a vector of the same length as the input feature x, where each

dimension of $f_{\text{loc}}(x)$ indicates the contribution of each feature x_i within the input sample x to the prediction of the model $f(x)$. Note that not all local explanation algorithms can be used here, such as LIME [49] and SHAP [42]. Instead, $f_{\text{loc}}(x)$ should be end-to-end differentiable, so that backpropogation can be used to update model parameters. The motivation is to suppress the model's attention on fairness-sensitive features and instead encourage the model to focus more on task-relevant features. An example is toxic classification, where explanations at the feature level (obtained from the integrated gradient) are encouraged to be consistent with the rationale of domain experts [39]. More specifically, the models would predict every sentence containing the word 'gay' as a toxic comment. Similarly, sentences with word 'jew' and 'black' would be given negative prediction for the sentiment analysis task. The rationale would specify the list of fairness-sensitive features, and the regularization would penalize the model's attention on these sensitive features. Although this method is effective in terms of bias mitigation, it requires sensitive feature annotations. This presents some challenges. First, annotating the exclusive list of sensitive features is expensive and time consuming. Furthermore, it is not clear which subset of features is sensitive to fairness, especially for real-world applications.

Debiasing Entire Model: Fairness Metric Regularization. Second, in addition to regularization of explanations, another most commonly used debiasing method is to directly add fairness metrics to the loss function. We will use the demographic parity metric as an illustration below.

$$L(\theta,x,y,a) = \underbrace{d_1(y,\hat{y})}_{\text{Prediction}} + \underbrace{\lambda_1 \mathcal{F}_{DP}}_{\text{Fairness}} + \underbrace{\lambda_2 \mathcal{R}(\theta)}_{\text{Requlatizer}}, \tag{13}$$

where $\mathcal{F}_{DP}$ is the demographic parity metric. A larger λ_1 will impose stronger regularization, at the expense of a greater accuracy trade-off. Note that fairness metrics such as $\mathcal{F}_{DP}$ are not end-to-end differentials. Therefore, we need to relax the calculation of the fairness metric in the following format [43, 9].

$$\mathcal{F}_{DP} = \left|\mathbb{E}_{x\sim P_0} f(x) - \mathbb{E}_{x\sim P_1} f(x)\right| \tag{14}$$

where $P_0 = P(\cdot \mid a = 0)$ and $P_1 = P(\cdot \mid a = 1)$, denoting the distribution of two protected groups. We can calculate the relaxed fairness metric by sampling two protected groups from a batch of data during the implementation stage. Similarly, we can regularize the equality of odds metric.

$$L(\theta,x,y,a) = \underbrace{d_1(y,\hat{y})}_{\text{Prediction}} + \underbrace{\lambda_1 \mathcal{F}_{EOO}}_{\text{Fairness}} + \underbrace{\lambda_2 \mathcal{R}(\theta)}_{\text{Requlatizer}}. \tag{15}$$

The fairness metric $\mathcal{F}_{EOO}$ also does not have end-to-end differential ability. It can be relaxed in the following formulation [43, 9].

$$\mathcal{F}_{EOO} = \sum_{y\in\{0,1\}} \left|\mathbb{E}_{x\sim P_0^y} f(x) - \mathbb{E}_{x\sim P_1^y} f(x)\right| \tag{16}$$

where $P_0^y = P(\cdot \mid a = 0, Y = y), y \in \{0,1\}$ and $P_1^y = P(\cdot \mid a = 1, Y = y), y \in \{0,1\}$. They are also calculated by using a batch of data. It should be noted that optimizing either the demographic parity or the equality of odds metric may only improve the specific metric rather than simultaneously improving all fairness metrics.

Debiasing Entire Model: Implicit Regularization. Third, we do not need to explicitly add the fairness term, as used in Equation 12. As such, the debiasing algorithm takes the following formulation.

$$L(\theta, x, y, a) = \underbrace{d_1(y, \hat{y})}_{\text{Prediction}} + \underbrace{\lambda_2 \mathcal{R}(\theta)}_{\text{Requlrizer}} . \tag{17}$$

Here, the regularizer itself can achieve the debiasing effect. An implementation of this debiasing method is to use weight decay [46].

$$L(\theta, x, y, a) = \underbrace{d_1(y, \hat{y})}_{\text{Prediction}} + \underbrace{\lambda_2 \|\theta\|_2^2}_{\text{Weight decay}} . \tag{18}$$

In addition, we can implement this using spectral decoupling [46].

$$L(\theta, x, y, a) = \underbrace{d_1(y, \hat{y})}_{\text{Prediction}} + \underbrace{\lambda_2 \|\hat{y}\|_2^2}_{\text{Spectral decoupling}} . \tag{19}$$

Both weight decay and spectral decoupling are motivated by the perspective of shortcut learning [25]. Specifically, models trained with cross-entropy loss tend to rely on a small subset of features (in this case, fairness-sensitive features) for prediction and fail to learn other predictive features (here, task-relevant features). This phenomenon is named *gradient starvation* [46]. Both weight decay and spectral decoupling can not stop the model from learning fairness-sensitive features, and instead encourage the models to suppress their attention on fairness-sensitive features. In other words, they are useful primarily for altering the model's classification head.

Comparisons of the Three Paradigms. We provide the pros and cons of the three mentioned debiasing paradigms, that is, debiasing the representation, the classification head, and the entire model.

- Recent research suggests that it can be challenging to eliminate biased information from representations. Experimental findings have revealed two drawbacks of adversarial training. At first, it might also remove certain information relevant to tasks. Therefore, there would be a significant trade-off between fairness and utility. Second, there may be significant variation between different runs, as adversarial training is not stable in training.
- The more precise classification boundary is the main advantage of debiasing the classification head. Assume that there are both features relevant to the task and features sensitive to fairness in the learned deep representations. The implicit result of this type of debiasing is to enable the model to shift its focus from fairness-sensitive features to task-relevant features.

- Debiasing the entire model can simultaneously adjust the encoder and the classification head. However, the relative role of the encoder and the classification head in terms of debiasing the entire model is unclear. It is an interesting direction to diagnose the debiased model to understand whether the improvement is primarily due to the debiased representation or the refined decision boundary.

2.2.2 Prediction Quality Disparity

Many of the mitigation methods share the same philosophy as methods that address the problem of long-tailed classification.

One of the most representative methods is reweighting [38]. It contains two steps, where the goal of the first step is to identify the unprivileged groups. The identification model is trained in only a few epochs. The assumption is that the biased model would give low prediction accuracy for unprivileged groups.

$$E = \left\{ (x_i, y_i) \text{ s.t. } \hat{f}_{\text{id}}(x_i) \neq y_i \right\} \tag{20}$$

In the second step, the debiasing algorithm would give higher weights to these samples that correspond to unprivileged groups.

$$J_{\text{up-ERM}}(\theta, E) = \left(\lambda_{\text{up}} \sum_{(x,y)\in E} \ell(x, y; \theta) + \sum_{(x,y)\notin E} \ell(x, y; \theta) \right) \tag{21}$$

where λ_{up} is the weight of the reweighting. It is important to note that in this debiasing approach, we assume that we do not have access to the protected attribute annotations. In contrast, if there are annotations for the protected attribute, we can directly assign higher training weights to training samples from the unprivileged group.

Another effective method is self-supervised pre-training (see Figure 3) [45]. This is because, if the model is trained from scratch, the underprivileged group might not be adequately represented in the labeled training dataset due to factors such as sampling bias. The quality of the training data for unprivileged groups is much lower than that of the privileged groups; either the number is much fewer or the quality of the label is lower. As such, the machine learning model was unable to learn high-quality representations for the unprivileged group. To bridge this gap, the objective of pre-training is to learn good representations from unlabeled large-scale data (see Figure 3 (b)). This pre-trained model can then be finetuned in the labeled data for the downstream task.

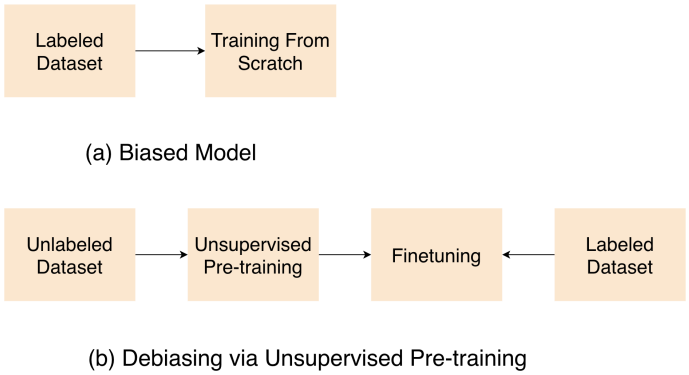

Fig. 3 Prediction quality disparity debiasing through self-supervised pre-training. The main bottleneck of quality disparity is the poor representation of the unprivileged group. Unsupervised pre-training could help learn high quality representations.

2.3 Post-processing Methods

This family of techniques performs the mitigation after training the machine learning model. We currently have a trained model. Mitigation can be used to alter the model's parameters and architectural structures or its prediction probability distribution.

2.3.1 Prediction Outcome Discrimination

Two types of mitigation methods include model prediction calibration and sensitive neuron pruning. It should be noted that the methods in the first category can be utilized in any machine learning model, whereas the methods in the second group can only be employed in DNN-based models.

First, we can calibrate the prediction of the models during the inference phase. The probability of prediction of the model could indicate the confidence of the model. For example, we can modify the predicted labels using a scheme that solves the linear program to optimize the classifier for better equalized odds performance [47]. In addition, the corpus-level constraint can be added to the existing prediction model to encourage the output of the model to follow a desirable distribution [67].

Second, after the model has been trained, we can identify neurons that encode the concept related to protected attributes. These neurons can be named troubling neurons. Then we can turn off the activation of neurons or remove the neurons. For example, one work proposes to identify parameters that are not important for unprivileged groups but important for privileged groups [64].

$$\min \Delta E_{a=0}(\theta), \max \Delta E_{a=1}(\theta) \tag{22}$$

which is then transformed into a single objective as follows:

$$\min(\Delta E_{a=0}(\theta) - \beta \Delta E_{a=1}(\theta)), \tag{23}$$

where β is used to control the trade-off between the importance calculation for the unprivileged and the privileged groups. Furthermore, ΔE is calculated using the second derivative of the parameters to quantify the increase in the prediction error after the model is pruned. They first sample mini-batches from the unprivileged and privileged groups, respectively. Based on the calculation, they remove a small ratio of parameters with the smallest values corresponding to Equation 23. This process is repeated for certain iterations until the target fairness measurement is reached. Note that the method based on neuron pruning could sacrifice model prediction performance. The possible reason is that a neuron could capture multiple concepts [13]. In other words, pruned neurons could also capture relevant information about the task. As a result, this pruning might dramatically affect task performance.

2.3.2 Prediction Quality Disparity

As introduced earlier, the prediction quality disparity is mainly caused by the poor representation learned by the machine learning models. Thus post-processing based methods might not be as effective as pre-processing methods and in-processing methods. However, we can improve prediction quality parity by learning from approaches that address the long-tailed classification problem, e.g., by modifying logit values during the inference stage [30].

3 Opportunities and Challenges

The current practice in machine learning research is to divide existing data into training, validation, and test sets. The models are tested using some benchmark dataset such as Adult Census Income [35], German Credit, ProPublica Recidivism (COMPAS) and CelebA [41]. We call it *laboratory setting*. It is time for researchers and developers from academia and industry to consider the implications of debiasing algorithms beyond the laboratory setting, i.e., *real world scenarios*. In particular, the laboratory setting does not take into account the complexity of real-world applications. As we move from the laboratory to real-world scenarios, there are some research challenges and opportunities that deserve the community's attention.

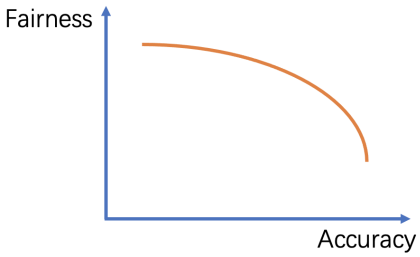

Fig. 4 Current research usually assumes that there is fairness and accuracy trade-off. When the fairness metric performance increases, the accuracy performance will drop.

3.1 Fairness and Utility Trade-off

For prediction outcome discrimination, the published literature generally assumes that there is a trade-off between fairness and utility (see Figure 4). This holds true for the laboratory setting, where the three subsets (i.e. training, validation, and test sets) are independent and identically distributed. It is 'beneficial' for all three subsets that the models overassociate the fairness-sensitive features with class labels. As such, the debiasing methods that decorrelate the undesirable correlations could sacrifice the models' performance, resulting in the fairness and utility trade-off. This trade-off might not hold true in real-world situations, where there may be a domain shift and the test data may be out-of-distribution (OOD) compared to the training data. In that case, it is likely that debiasing techniques could boost both fairness and utility if the test data originate from real-world applications [57]. Additionally, by improving fairness through aleatoric uncertainty [56], the fairness-utility trade-off could be optimized. This is a challenging topic and deserves further research from the community.

3.2 Intersectional Fairness

Due to restrictions in the laboratory environment, intersectional fairness is comparatively understudied in the existing literature, which often focuses on minimizing bias for one single sensitive feature. This simplified paradigm fails to recognize the intersectional harms resulting from interacting systems of oppression [60, 27]. Intersectionality refers to the ways in which discrimination manifesting in sociotechnical systems can be hidden when evaluating fairness on distinct sensitive attributes, without considering how identities and experiences might intersect in unique ways [10]. Intersectional fairness requires that the model prediction is roughly the same for all groups defined by different intersections of protected attributes. It is also referred to as compositional fairness in the literature as one person generally corresponds to multiple sensitive attributes in real-world applications, for example, a 66-year-old

Hispanic female. Recent years have witnessed a growing interest in intersectionality in ML fairness. For example, early works such as [32] proposed multi-attribute fairness definitions to prevent fairness gerrymandering at the intersections of protected groups. It downweighs underrepresented groups, which arguably should be the focus of intersectional fairness. Differential fairness [21], motivated by differential privacy, addresses this limitation. However, most current works consider intersectionality simply as a multi-attribute problem, failing to critically engage with the uniqueness and complexity of intersectionality such as the cultural differences, history, and context. A potential solution is to use geometry-based ML approaches to identify intersectional groups and mitigate intersectional bias in the *latent space* [7]. The predominant way to define intersectional fairness in the literature also overemphasizes the attributes, leading to problems such as infinite regress and reinscription of fairness gerrymandering [36]. A related yet different task is to jointly debias for multiple protected attributes [8]. If the mitigation strategy only addresses one kind of bias, such as gender bias, models can still exhibit other types of bias, such as racial and age bias. Even worse, the reduction of one bias might amplify the models' reliance on other biases for prediction [6, 8]. As a result, the models still suffer significant discrimination if applied to real-world applications. More research is needed to address the intersectional fairness problem [4, 60, 7].

3.3 Improving Fairness Without Demographics

Most existing methods assume that protected attribute annotations (i.e., demographic information) are available in the training set and use them to design mitigation methods. However, there are various kinds of problem with these protected attribute annotations in real-world scenarios. For example, the annotations are not available or are only partially available. Even for those with annotations, the protected attributes could be wrong and might not reflect the real information of the users. This could happen mainly for regulatory and privacy reasons. For some applications, it is forbidden by law to collect sensitive information from end users. Furthermore, some users may not be willing to share their information due to privacy concerns. Therefore, some mitigation methods propose to design mitigation frameworks without using sensitive information [14]. Typical solutions include 1) generating proxy annotations based on auxiliary tools and 2) active learning based sampling schemes to reduce the annotation efforts. However, the experimental results indicate that mitigation algorithms without using ground-truth annotations might achieve a worse fairness and accuracy trade-off. Therefore, more research is required from the community to develop stronger mitigation algorithms without relying on ground-truth protected attribute annotations.

3.4 Long-term Fairness

The current literature generally focuses on static or one-shot settings that care only about the static benefit of debiasing methods [40]. The key reason for debiasing methods is to reduce discrimination against unprivileged groups. However, recent research suggests that debiasing methods may ultimately harm the profits of disadvantaged groups after a certain period of time. As such, it is desirable to investigate the long-term fairness of debiasing algorithms [24]. Specifically, as debiasing algorithms are deployed in real-world applications, what is the performance of different demographic groups after a couple of years? However, this is a challenging topic. On the one hand, this is different from traditional machine learning practice in using static metrics to evaluate fairness performance. We need to design new metrics to access the long-term effect of fairness. Additionally, we often do not have data that span a significant amount of time as a result of the restricted available data in the laboratory setting. Existing work conducts studies through simulations [11], which may not accurately reflect real-world scenarios.

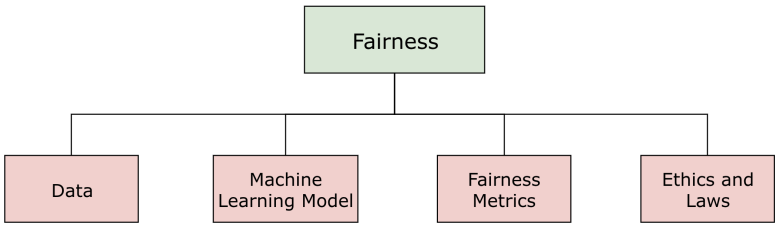

Fig. 5 Four fairness levels. In this work, we mainly discussed the fairness problem from the computational perspective. Beyond data bias and algorithmic bias, the definition of fairness metrics is also an crucial topic. To define a suitable metric, we need to consider the ethical and law requirements, and also need to take the domain knowledge from the experts into consideration.

3.5 Fairness Metrics

It is a challenging task to assess whether a machine learning model has achieved a fair result. There are two specific questions that need to be answered. First, what metrics/measurements should we use to evaluate fairness performance? It should be noted that different fairness metrics may be incompatible [48]. For example, a machine learning model may be fair under the demographic parity metric, but it would achieve poor performance under the equality of odds metric. Second, after selecting one specific fairness metric, how can we guarantee that a machine learning model is fair? Does fairness mean achieving parity or reaching a certain preference [48]? In Section 1.2, we mention that the performance of EOP and EOO should be as close

to 0 as possible. This is called parity-based metrics and is typically used in the literature. In real-world applications, the desirable gap is not as small as possible. For example, there is the 80% rule if we calculate the demographic parity ratio. As long as the ratio between the unprivileged group and the privileged group is greater than 80%, we would consider the model to be a fair model. However, either the parity-based requirement or the 80% rule are general rules, which might not be applicable for a specific application.

To address these two questions, we must take into account factors such as ethical criteria, law requirements, and the characteristics of the task (see Figure 5). In particular, experts in the domain of specific applications should be involved in defining the desirable performance that a fair model should achieve. In addition, data scientists, decision makers, and others affected by the use of the model are also suggested to have a thorough discussion [50]. Eventually, we can make a more informed decision about which fairness metrics to use.

4 Connections with Other Directions of Trustworthy AI

Algorithmic fairness, as a subfield of trustworthy AI, has many interdisciplinary research connections with other directions of trustworthy AI, such as explainability, privacy, and efficiency.

4.1 Connection with Explainability

Explainability can be used as an effective tool to shed light on the decision-making process of machine learning models. As such, explainability can be used to detect biases in models and incorporated into the model training process to improve the fairness of machine learning models [15].

- *Detecting Bias in Biased Models.* Machine learning models often rely on fairness-sensitive features for prediction. For DNNs, the learned deep representations could be biased by capturing bias information. As such, machine learning explainability can be used as a diagnostic tool to provide a thorough understanding of the causes of model bias. On the one hand, post-hoc local explainability can be employed to locate these features sensitive to fairness. On the other hand, post-hoc global explainability can be applied to detect when biased information has been captured in deep representations. A typical example is the concept activation vector (CAV) method, which is a concept-based model interpretability algorithm [33]. Consider the recruitment algorithm as an illustration. We can use CAV to diagnose the model to determine whether gender information has been captured in the intermediate representations of the DNN models.
- *Improve Fairness.* The key idea is to regularize the attention of machine learning models, such as those introduced in Section 2.2.1. Specifically, the explanation

algorithm will be incorporated into the overall loss function to regularize the models' training. For example, for the image caption task in the computer vision domain, visual attention loss is proposed to provide guidance on the model's attention, with the aim of encouraging the model to rely on correct visual evidence for prediction [57]. Experimental analysis indicates that the proposed regularization approach can dramatically minimize gender prediction errors while maintaining competitive caption quality.

- *Understanding the Debiased Models.* Despite the fact that numerous debiasing techniques have produced encouraging mitigating results, it is still unknown what caused the improvement. Debiased models can be examined in this situation using explainability methods. For example, consider the debiasing algorithms that add regularization to the entire model as introduced in previous sections. We can use explainability to determine whether improved performance is due to debiased representations or the refined classification head. This insight obtained would enable the community to create more effective debiasing algorithms.

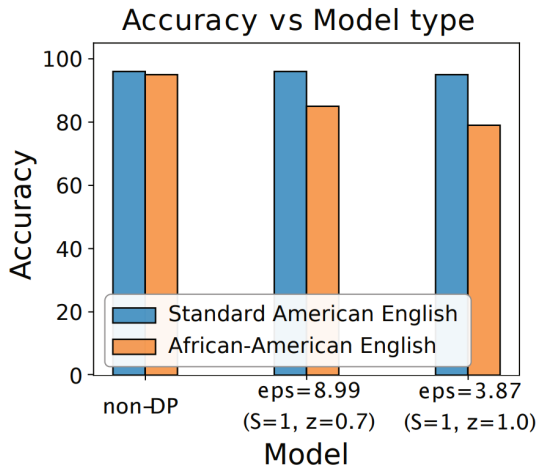

Fig. 6 Trade-off between differential privacy and fairness. Image is taken from Bagdasaryan et al. [1]. There is a certain unfairness for the original non-DP model. Privacy constraints would exacerbate model bias, via increasing the accuracy gap.

4.2 Connection with Privacy

The goal of differential privacy is to prevent the model from releasing sensitive information about individuals. Despite the fact that differential privacy could help to ensure privacy, it might also contribute errors to the task's outputs. It is shown that these errors may have distinct effects on various populations, and thus differential privacy might exacerbate model bias in many applications [20]. An example is from the sentiment classification task [1], as shown in Figure 6. Before applying the differential privacy algorithm, there is little accuracy disparity between standard American English and African American English. On the contrary, after imposing differential privacy constraints, there exist more than 10% accuracy gaps between these two groups. The overall accuracy drop comes mainly from the underrepresented African-American group. Furthermore, as privacy increases, the disparity between two groups also increases. Therefore, it is desirable to design algorithms that could simultaneously improve privacy and fairness.

4.3 Connection with Efficiency

Nowadays, DNN models are being increasingly used in real-world applications with latency and capacity constraints, such as in edge devices. Toward this end, there is a need to compress large DNN models into smaller ones using compression techniques such as knowledge distillation, pruning, matrix decomposition, and quantization. The current literature focuses on evaluation schemes that calculate overall accuracy and asserts that model performance can be preserved during compression. However, compressed models have disproportionately high errors in a small subset of samples [31]. This category includes disadvantaged and underrepresented groups such as women and African Americans. In other words, compression could amplify the model bias. For this reason, it is desirable to develop more equitable compression strategies that do not undermine the benefit of the model. On the plus side, this finding indicates that compression can automatically expose more difficult examples and thus could offer proxy annotations on the protected attributes that are disproportionately affected by compression. Equipped with these proxy annotations, we can design better mitigation solutions even without ground-truth annotations for the protected attribute.

5 Conclusions

In this chapter, we provide a comprehensive review of the current state of knowledge in this critical area of fairness in machine learning. First, we divide algorithmic fairness into two broad categories: prediction outcome discrimination and prediction quality parity. Based on this categorization, we introduce some typical fairness met-

rics. Second, we have discussed representative bias mitigation algorithms, from pre-processing, in-processing and post-processing three perspectives. Lastly, we further introduce research challenges and the connection of fairness in machine learning with other sub-areas of trustworthy AI.

References

[1] E. Bagdasaryan, O. Poursaeed, and V. Shmatikov. Differential privacy has disparate impact on model accuracy. *Advances in neural information processing systems*, 32, 2019.

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

[3] S. Bird, M. Dudík, R. Edgar, B. Horn, R. Lutz, V. Milan, M. Sameki, H. Wallach, and K. Walker. Fairlearn: A toolkit for assessing and improving fairness in ai. *Microsoft, Tech. Rep. MSR-TR-2020-32*, 2020.

[4] A. J. Bose and W. Hamilton. Compositional fairness constraints for graph embeddings. *International Conference on Machine Learning (ICML)*, 2019.

[5] J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *ACM Conference on Fairness, Accountability and Transparency (FAT*)*, pages 77–91, 2018.

[6] L. Cheng, S. Ge, and H. Liu. Toward understanding bias correlations for mitigation in nlp. *arXiv preprint arXiv:2205.12391*, 2022.

[7] L. Cheng, N. Kim, and H. Liu. Debiasing word embeddings with nonlinear geometry. In *COLING*, 2022.

[8] L. Cheng, A. Mosallanezhad, Y. N. Silva, D. L. Hall, and H. Liu. Bias mitigation for toxicity detection via sequential decisions. In *SIGIR*, pages 1750–1760, 2022.

[9] C.-Y. Chuang and Y. Mroueh. Fair mixup: Fairness via interpolation. *International Conference on Learning Representations (ICLR)*, 2021.

[10] K. Crenshaw. Demarginalizing the intersection of race and sex: A black feminist critique of antidiscrimination doctrine, feminist theory and antiracist politics. *u. Chi. Legal f.*, page 139, 1989.

[11] A. D'Amour, H. Srinivasan, J. Atwood, P. Baljekar, D. Sculley, and Y. Halpern. Fairness is not static: deeper understanding of long term fairness via simulation studies. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 525–534, 2020.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

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

[14] M. Du, S. Mukherjee, G. Wang, R. Tang, A. Awadallah, and X. Hu. Fairness via representation neutralization. *Advances in Neural Information Processing Systems (NeurIPS)*, 34, 2021.

[15] M. Du, F. Yang, N. Zou, and X. Hu. Fairness in deep learning: A computational perspective. *IEEE Intelligent Systems*, 2020.

[16] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, 2012.

[17] Y. Elazar and Y. Goldberg. Adversarial removal of demographic attributes from text data. *2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

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

[19] Q. Feng, M. Du, N. Zou, and X. Hu. Fair machine learning in healthcare: A review. *arXiv preprint arXiv:2206.14397*, 2022.

[20] F. Fioretto, C. Tran, P. Van Hentenryck, and K. Zhu. Differential privacy and fairness in decisions and learning tasks: A survey. *arXiv preprint arXiv:2202.08187*, 2022.

[21] J. R. Foulds, R. Islam, K. N. Keya, and S. Pan. An intersectional definition of fairness. In *ICDE*, pages 1918–1921. IEEE, 2020.

[22] M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan. Synthetic data augmentation using gan for improved liver lesion classification. In *IEEE international symposium on biomedical imaging (ISBI)*, 2018.

[23] P. Gajane and M. Pechenizkiy. On formalizing fairness in prediction with machine learning. *Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*, 2018.

[24] Y. Ge, S. Liu, R. Gao, Y. Xian, Y. Li, X. Zhao, C. Pei, F. Sun, J. Ge, W. Ou, et al. Towards long-term fairness in recommendation. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*, pages 445–453, 2021.

[25] R. Geirhos, J.-H. Jacobsen, C. Michaelis, R. Zemel, W. Brendel, M. Bethge, and F. A. Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[26] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2018.

[27] U. Gohar and L. Cheng. A survey on intersectional fairness in machine learning: Notions, mitigation, and challenges. In *IJCAI*, 2023.

[28] M. Hardt, E. Price, N. Srebro, et al. Equality of opportunity in supervised learning. In *Advances in neural information processing systems (NeurIPS)*, 2016.

[29] K. Hermann, T. Chen, and S. Kornblith. The origins and prevalence of texture bias in convolutional neural networks. *Advances in Neural Information Processing Systems*, 33:19000–19015, 2020.

[30] Y. Hong, S. Han, K. Choi, S. Seo, B. Kim, and B. Chang. Disentangling label distribution for long-tailed visual recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6626–6636, 2021.

[31] S. Hooker, N. Moorosi, G. Clark, S. Bengio, and E. Denton. Characterising bias in compressed models. *arXiv preprint arXiv:2010.03058*, 2020.

[32] M. Kearns, S. Neel, A. Roth, and Z. S. Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *ICML*, pages 2564–2572. PMLR, 2018.

[33] B. Kim, M. Wattenberg, J. Gilmer, C. Cai, J. Wexler, F. Viegas, and R. Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). *International Conference on Machine Learning (ICML)*, 2018.

[34] A. Koenecke, A. Nam, E. Lake, J. Nudell, M. Quartey, Z. Mengesha, C. Toups, J. R. Rickford, D. Jurafsky, and S. Goel. Racial disparities in automated speech recognition. *Proceedings of the National Academy of Sciences*, 117(14):7684–7689, 2020.

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

[36] Y. Kong. Are "intersectionally fair" ai algorithms really fair to women of color? a philosophical analysis. In *2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 485–494, 2022.

[37] P. Lahoti, A. Beutel, J. Chen, K. Lee, F. Prost, N. Thain, X. Wang, and E. Chi. Fairness without demographics through adversarially reweighted learning. *Advances in neural information processing systems*, 33:728–740, 2020.

[38] E. Z. Liu, B. Haghgoo, A. S. Chen, A. Raghunathan, P. W. Koh, S. Sagawa, P. Liang, and C. Finn. Just train twice: Improving group robustness without training group information. In *International Conference on Machine Learning*, pages 6781–6792. PMLR, 2021.

[39] F. Liu and B. Avci. Incorporating priors with feature attribution on text classification. *57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.

[40] L. T. Liu, S. Dean, E. Rolf, M. Simchowitz, and M. Hardt. Delayed impact of fair machine learning. In *International Conference on Machine Learning*, pages 3150–3158. PMLR, 2018.

[41] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, 2015.

[42] S. M. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4768–4777, 2017.

[43] D. Madras, E. Creager, T. Pitassi, and R. Zemel. Learning adversarially fair and transferable representations. In *International Conference on Machine Learning*, pages 3384–3393. PMLR, 2018.

[44] N. Mehrabi, F. Morstatter, N. Saxena, K. Lerman, and A. Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 2021.

[45] A. Newell and J. Deng. How useful is self-supervised pretraining for visual tasks? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7345–7354, 2020.

[46] M. Pezeshki, O. Kaba, Y. Bengio, A. C. Courville, D. Precup, and G. Lajoie. Gradient starvation: A learning proclivity in neural networks. *Advances in Neural Information Processing Systems*, 34:1256–1272, 2021.

[47] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger. On fairness and calibration. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 5680–5689, 2017.

[48] A. Rajkomar, M. Hardt, M. D. Howell, G. Corrado, and M. H. Chin. Ensuring fairness in machine learning to advance health equity. *Annals of internal medicine*, 169(12):866–872, 2018.

[49] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, 2016.

[50] K. T. Rodolfa, P. Saleiro, and R. Ghani. Bias and fairness. In *Big data and social science*, pages 281–312. Chapman and Hall/CRC, 2020.

[51] D. S. Shah, H. A. Schwartz, and D. Hovy. Predictive biases in natural language processing models: A conceptual framework and overview. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5248–5264, 2020.

[52] H. Shah, K. Tamuly, A. Raghunathan, P. Jain, and P. Netrapalli. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems*, 33:9573–9585, 2020.

[53] K. Shailaja, B. Seetharamulu, and M. Jabbar. Machine learning in healthcare: A review. In *2018 Second international conference on electronics, communication and aerospace technology (ICECA)*, pages 910–914. IEEE, 2018.

[54] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[55] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. *International Conference on Machine Learning (ICML)*, 2017.

[56] A. Tahir, L. Cheng, and H. Liu. Fairness through aleatoric uncertainty. In *CIKM*, 2023.

[57] R. Tang, M. Du, Y. Li, Z. Liu, N. Zou, and X. Hu. Mitigating gender bias in captioning systems. In *Proceedings of the Web Conference 2021*, pages 633–645, 2021.

[58] C. Wadsworth, F. Vera, and C. Piech. Achieving fairness through adversarial learning: an application to recidivism prediction. *Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*, 2018.

[59] M. Wan, D. Zha, N. Liu, and N. Zou. Modeling techniques for machine learning fairness: A survey. *arXiv preprint arXiv:2111.03015*, 2021.

[60] A. Wang, V. V. Ramaswamy, and O. Russakovsky. Towards intersectionality in machine learning: Including more identities, handling underrepresentation, and performing evaluation. *arXiv preprint arXiv:2205.04610*, 2022.

[61] A. Wang and O. Russakovsky. Directional bias amplification. In *International Conference on Machine Learning*, pages 10882–10893. PMLR, 2021.

[62] T. Wang, J. Zhao, M. Yatskar, K.-W. Chang, and V. Ordonez. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. *International Conference on Computer Vision (ICCV)*, 2019.

[63] F. Wu, M. Du, C. Fan, R. Tang, Y. Yang, A. Mostafavi, and X. Hu. Understanding social biases behind location names in contextual word embedding models. *IEEE Transactions on Computational Social Systems*, 9(2):458–468, 2021.

[64] Y. Wu, D. Zeng, X. Xu, Y. Shi, and J. Hu. Fairprune: Achieving fairness through pruning for dermatological disease diagnosis. *arXiv preprint arXiv:2203.02110*, 2022.

[65] M. B. Zafar, I. Valera, M. Gomez Rodriguez, and K. P. Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *International Conference on World Wide Web (WWW)*, 2017.

[66] B. H. Zhang, B. Lemoine, and M. Mitchell. Mitigating unwanted biases with adversarial learning. In *AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society (AIES)*, 2018.

[67] J. Zhao, T. Wang, M. Yatskar, V. Ordonez, and K.-W. Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. *2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017.

[68] J. Zhao, T. Wang, M. Yatskar, V. Ordonez, and K.-W. Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.