

MAAD PRIVATE: MULTI-ATTRIBUTE ADVERSARIAL DEBIASING WITH DIFFERENTIAL PRIVACY

Anonymous authors

Paper under double-blind review

ABSTRACT

Balancing the trade-offs between algorithmic fairness, individual privacy, and model utility, is pivotal for the advancement of ethical artificial intelligence. In this work, we explore fair classification through the lens of differential privacy. We present an enhancement to the adversarial debiasing approach, enabling it to account for multiple sensitive attributes while upholding a privacy-conscious learning paradigm. Empirical results from two tabular datasets and a natural language dataset demonstrate our model’s ability to concurrently debias up to four sensitive attributes and meet various fairness criteria, within the constraints of differential privacy.

1 INTRODUCTION

With the increasing application of ML in domains like healthcare and law enforcement, the processing of sensitive data is becoming commonplace. In order to avoid exhibiting biases for specific demographic groups Mehrabi et al. (2021) Pessach & Shmueli (2022), memorizing of sensitive data Carlini et al. (2019), revealing of identity Hu et al. (2022); Shokri et al. (2017), or inferring personal attributes Ganju et al. (2018); Parisot et al. (2021), developing and evaluating ML models requires not only attention to performance, but also to fairness and privacy. Ensuring fairness and privacy in algorithms is intricate due to the varied intervention points of each. For fairness, Caton & Haas (2020) distinguishes between pre-processing interventions, which aim to correct biases in data before learning, in-processing interventions, which occur during learning, and post-processing interventions, which happen after learning. For differential privacy, similarly, data are protected by either adding noise in the pre-processing phase, during learning itself, or through post-processing Dwork (2008); Ji et al. (2014); Petti & Flaxman (2019); McSherry & Talwar (2007). To prevent one taking precedence over the other (see below), fairness and privacy need to be considered concurrently. One of the most promising ways of doing so, is by injecting differential privacy (DP) into existing fairness algorithms Jagielski et al. (2019). One shortcoming of this approach is that it only considers single sensitive attributes, despite real-world scenarios often involving multiple attributes. Another limitation is that the classifier is trained without a differentially private optimizer, which can allow inferring values for attributes which are irrelevant for fairness but that are sensitive from a privacy perspective.

In our study, we employ an in-processing fairness method, using gradient-based learning algorithms integrated with differential privacy Abadi et al. (2016). Gradient-driven models, which iteratively adjust parameters based on computed gradients, become particularly cautious about revealing individual data under differential privacy constraints. Integrating fairness and privacy during training ensures simultaneous optimization for both, eliminating the potential tug-of-war that might occur when trying to bolt on fairness or privacy considerations after or before the fact. In contrast, pre-processing methods might distort data when paired with differential privacy, impacting fairness and privacy. Meanwhile, post-processing methods only adjust outputs, leaving potential biases in the core model, which differential privacy could further exacerbate.

2 RELATED WORK

2.1 ADVERSARIAL DEBIASING

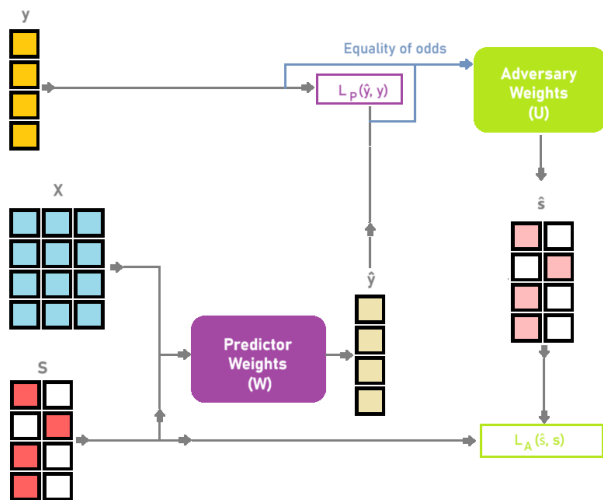
Adversarial debiasing Zhang et al. (2018) is an in-processing technique in machine learning. It aims to mitigate the entrenchment of data biases leading to unfair classifications (see 4.3). Central to its design is the application of gradient reversal, which involves modifying the gradients of a particular sub-network during training. This encourages the model to learn domain-invariant features or to mitigate bias. In other words, gradients originating from the adversarial component, which aims to predict sensitive attributes, are systematically inverted during back-propagation. If the adversary succeeds in its task, it indicates a biased classifier. However, by inverting these gradients, the classifier’s reliance on both explicit sensitive attributes and their correlations with other features (like the association between marital status and sex) is diminished, ensuring a more unbiased classification. In doing so, this tactic disrupts the model’s propensity to lean on sensitive attributes, culminating in more equitable predictions by strategically obfuscating biases.

Adversarial debiasing is a versatile framework suitable for multiple fairness definitions. By tailoring the adversarial network’s inputs, one can target fairness metrics like demographic parity, equalized odds, or equality of opportunity.

- **Demographic Parity:** Achieved when predictions \hat{Y} and the protected attribute S are independent. For enforcement, the adversary should access only the predicted value \hat{Y} , trying to infer S from it.
- **Equality of Odds:** Satisfied when \hat{Y} and S are conditionally independent given Y . The adversary receives both the predicted label \hat{Y} and the actual label Y .
- **Equality of Opportunity:** Met for a specific class y when \hat{Y} and Z are independent given $Y = y$. Here, the adversary’s data only includes instances where $Y = y$.

In the current work, we adopt Equality of Odds (Hardt et al., 2016) as a definition of fairness, which conditionally assesses fairness based on ground truth and aims for error parity among sensitive groups.

Figure 1: **Enforcing equality of odds via Adversarial Debasing** : X represents all non-sensitive input features for the predictor network, S denotes all sensitive feature inputs, and y is the binary target. Parameters of the predictor and adversary networks are denoted by W and U . The loss function for the predictor and adversary are represented by $LP(\hat{y}, y)$ and $LA(\hat{s}, s)$ respectively



2.2 DIFFERENTIAL PRIVACY

Differential Privacy, introduced by Dwork (2006), sets a standard for measuring the disclosure of individual data during computations. A computation in the context of differential privacy is described as a mechanism M . This mechanism complies with (ϵ, δ) -differential privacy under specific conditions. For any potential output and for any pair of neighboring datasets D and D' differing by a single entry, the mechanism’s likelihood to produce that output remains largely consistent. This consistency is maintained irrespective of whether a particular individual’s data is included or excluded. Mechanism M satisfies ϵ differential privacy when the output probability for any two inputs differing by a single data point and generating diverse outcomes, is constrained within a factor of $\exp(\epsilon)$.

$$Pr[M(D) \in S] \leq \exp(\epsilon) \cdot Pr[M(D') \in S] + \delta \quad (1)$$

Differential privacy ensures outputs don’t reveal specific individual details, even with extra dataset knowledge. The parameter ϵ denotes privacy level; lower values mean stronger protection. However, strict adherence can reduce data utility, leading to ϵ - δ differential privacy. Here, a small constant, δ , denotes the maximum breach probability for a set ϵ , indicating the risk of revealing individual data. Typically, a δ less than $1/\text{dataset size}$ is sufficient Zhao et al. (2019). If δ is zero, it’s purely ϵ -differential privacy. Both parameters determine privacy level and output noise. Differential privacy allows combining different differentially private methods, maintaining its guarantees (compositionality) and remains unaffected even if the output of the mechanism undergoes further processing or analysis that is not dependent on the original data (post-processing immunity) Dwork et al. (2014).

When training certain machine learning algorithms, differential privacy can be implemented by adding noise to gradients each iteration, preventing over-fitting to individual data and preserving privacy. Techniques, such as gradient clipping and noise insertion, are used to manage cumulative noise and maintain the privacy budget. The Moments Accountant technique¹, introduced by Abadi et al. (2016), efficiently tracks privacy loss across iterations by evaluating the moments of the privacy loss variable, offering precise privacy cost estimation.

2.3 DEBIASING MULTIPLE SENSITIVE ATTRIBUTES

Recent literature offers adaptations to traditional debiasing techniques for managing multiple sensitive attributes. Notably, Kang et al. (2022) introduces an information-theoretic approach, InfoFair, which enforces statistical parity for multiple attributes via a vectorized sensitive attribute. This approach first transforms multiple sensitive attributes into one multi-dimensional feature, subsequently minimising both prediction error and sensitive attribute influence. The latter is achieved by limiting the mutual information between learning outcomes and the vectorized attributes, via a component in InfoFair’s loss function. While Kang et al. (2022) compared InfoFair to numerous multi-attribute debiasing methods, their work primarily highlights two results. First, InfoFair consistently reduces bias without significantly sacrificing base classification performance across various datasets. Second, other methods like LFR, Adversarial Debiasing, and FCFC might completely eliminate bias but do so with a significant performance cost, either labeling data uniformly or not surpassing InfoFair in certain settings.

2.4 DIFFERENTIALLY PRIVATE VERSIONS OF FAIRNESS ALGORITHMS

Jagielski et al. (2019) adapted two fairness algorithms using differential privacy. The first method is a post-processing fairness intervention based on Hardt et al. (2016). The approach begins with a potentially unfair classifier, Y_p . It refines fairness by blending Y_p with classifiers trained on protected attributes, with quantities $\hat{q}_{\hat{y}ay}$ ² acting as input to a linear program. To uphold differential privacy regarding these attributes, Y_p is first learned without considering protected attributes. Perturbation techniques are then applied to the $\hat{q}_{\hat{y}ay}$ values, ensuring differential privacy before inputting them into the linear program.

¹Please see TensorFlow Privacy library, Source: <https://github.com/tensorflow/privacy>

²This is the fraction of data points with $\hat{Y} = \hat{y}, A = a, Y = y$

The second, inspired by Agarwal et al. (2018), offers a privacy-adapted take on the oracle-efficient in-processing approach. This approach achieves optimal classification, sidestepping fairness constraints, often through basic learning heuristics. Central to this is a zero-sum game between a “Learner”, who chooses classifiers, and an “Auditor”³ overseeing fairness, a concept from Kearns et al. (2018). Their equilibrium is reached by the Auditor using gradient descent and the Learner employing cost-sensitive classification.

3 MULTI-ATTRIBUTE ADVERSARIAL DEBIASING

The approach we present in this paper modifies the implementation of Zhang et al. (2018), while building upon the work of Jagielski et al. (2019) in two important ways. First, while Jagielski et al. (2019) approach focuses on debiasing a single attribute (race), MAAD-private allows for concurrent debiasing of multiple sensitive features, mirroring real-world multi-bias data situations. Second, Jagielski et al. (2019)’s choice to add noise to the Auditor’s gradient does not yet account for the essential role of the Learner. Given that the Learner adapts to both sensitive and non-sensitive data, noise introduction is pivotal to prevent potential privacy leaks from the Learner’s precise gradient knowledge. Unlike the Auditor’s reactive stance, the Learner’s proactive function necessitates this early privacy integration. In essence, while both entities are crucial, the Learner’s gradient noise is vital for comprehensive privacy protection.

Furthermore, in addition to the multi-attribute debiasing presented by Kang et al. (2022), this work integrates differential private training and compares performance with non-private scenarios. We evaluate Adversarial debiasing within differential privacy, contrasting results against non-private and biased benchmarks. Another divergence from Kang et al. (2022) is our adoption of Equality of Odds Hardt et al. (2016) instead of Statistical Parity as a fairness metric. Equality of Odds captures both false positive and false negative rates, which are pivotal for sectors like credit assessment or recruitment, aligning with both legal and ethical norms. Finally, the current work expands earlier evaluation, which was limited to the communities and crime dataset Frank (2010), to multiple new datasets, including tabular and natural language domains.

4 METHODS

4.1 DATA

We use three datasets previously employed in the study of fair ML. Adult Kohavi (1996) and COMPAS Angwin et al. (2016) are traditional tabular datasets with *race* and *sex* as sensitive attributes. Additionally, we use a dataset with unstructured text entries derived from the multilingual twitter corpus collected by Huang et al. (2020) in the context of hate speech classification. For both the Adult and COMPAS datasets, we used standard pre-processing approaches provided by Bellamy et al. (2018). For the hate speech dataset, we tokenize the tweets and converted them into uniform-length integer sequences using padding and truncation⁴. Following the approach of Deriu et al. (2017); Bojanowski et al. (2017), these sequences were then transformed using the pre-trained word embeddings, specifically Google News Word2Vec embeddings Church (2017). This results in a dense representation for each tweet during training. For fairness evaluations, we use the demographic annotations provided by Huang et al. (2020), being *age*, *gender*, *ethnicity*, and *country*. These annotations were used to assess potential fairness violations after training the model.

4.2 MODELS

In all tabular datasets, the predictor network consists of a standard feed-forward Multi-Layer Perceptron (MLP) architecture with two hidden layers of 256 neurons and an output layer with a sigmoid activation function. This configuration allows for effective feature extraction and classification in the tabular domain. For the hate speech dataset, we use a transformer architecture to capture contextual

³Our work uses the terms “Predictor” and “Adversary” in lieu of “Learner” and “Auditor”, following the convention termed by Zhang et al. (2018)

⁴We use pre-processing functions provided by TensorFlow’s Keras library for this, Source: <https://github.com/keras-team/keras>

relationships within the text Vaswani et al. (2017). Our transformer architecture has four attention heads. Each feed-forward dimension within the transformer consists of 64 neurons, and the output layer uses a sigmoid activation, similar to the MLP architecture, given the binary classification task.

4.3 TRAINING REGIMEN

The architecture relies on two modules, the “predictor” and the “adversary”. The predictor model estimates the target Y from features X , refining its weights W via gradient descent to minimize the loss $LP(\hat{y}, y)$. This predictor’s output acts as the input for the adversary module, designed to predict sensitive attributes S . This setup mirrors the discriminator in generative adversarial architectures Goodfellow et al. (2014). The adversary has its loss function $LA(\hat{S}, S)$ and weight set, U . Note, \hat{Y} here refers to the network’s output layer, not the discrete prediction—like the sigmoid layer’s outcome in classification tasks. The pseudocode for training it is provided below.

Algorithm 1 MAAD-DP

Require: Training data $D_{\text{train}} = \{(x_i, y_i, z_i)\}_{i=1}^N$
Require: Learning rates α_c, α_a
Require: Adversary loss weight γ
Require: Noise multiplier and gradient norm bound

- 1: Initialize predictor P and adversary A , with weights W and U , respectively
- 2: **for** each epoch **do**
- 3: **for** each batch in D_{train} **do**
- 4: $P_y \leftarrow P(x)$
- 5: Compute loss L_p using \hat{y} and y
- 6: Predict sensitive attributes \hat{z} using \hat{y} and y
- 7: Compute loss L_a using \hat{z} and z
- 8: Compute gradients of L_a w.r.t W
- 9: Compute gradients of L_p w.r.t W
- 10: **for** each P grad **do**
- 11: Subtract projection onto normalized A gradients
- 12: Subtract $\gamma \times A$ gradients
- 13: **end for**
- 14: Clip P gradients by gradient norm bound
- 15: Add Gaussian noise scaled by multiplier to P gradients
- 16: Update W using noised clipped gradients
- 17: Compute gradient of L_a w.r.t U
- 18: Update U using these gradients
- 19: **end for**
- 20: **end for**

Ensure: Predictor P

Thus during training, the adversary adjusts weights U to minimize the LA loss using the gradient ∇ULA . Concurrently, the predictor’s weights W are updated as:

$$\nabla_W LP - \text{proj} \nabla_W LA \nabla_W LP - \gamma \nabla_W LA \quad (2)$$

To understand equation 2, each term can be broken down as the following:

- $\nabla_W LP$: represents the gradient of the loss with respect to the model parameters W for the primary task (e.g., the prediction task). This gradient provides the direction in which we need to adjust the model’s weights to minimize the prediction loss.
- $\nabla_W LA$: represents the gradient of the loss with respect to the model parameters W for the adversarial task. The adversarial task aims to predict a set of sensitive attributes (e.g., gender and race) from the model’s predictions, and we would like the model to perform poorly on this task to ensure it is not using the sensitive attribute to make its primary predictions.

- **proj**: This operation projects the gradient of the primary task onto the gradient of the adversarial task. This helps in ensuring that the update to the model’s weights doesn’t improve its performance on the adversarial task.
- γ : This is a hyper-parameter that controls the trade-off between the primary task and the adversarial task. A larger value of γ puts more emphasis on the adversarial task, thereby pushing the model to be more debiased.

In summary, the update rule adjusts the model’s weights to optimize its main function while minimizing dependence on sensitive attributes, using the gradient from the adversarial task. Although Zhang et al. (2018) focused on a single attribute, our method adapts the adversary to predict multiple attributes simultaneously. Each binary attribute gets its own sigmoid output, and the adversary’s loss is averaged over all these predictions. This average loss influences the gradient reversal, aiming for a representation devoid of sensitive information.

4.4 EXPERIMENTAL CONDITIONS

Experiments were conducted under four distinct training setups: a basic classification model without modifications; a second model with differential privacy optimization; a third model using adversarial gradient reversal for debiasing without differential privacy; and a fourth combining both differential privacy and adversarial debiasing. Each setup was evaluated 10 times on different dataset splits, with means and standard deviations presented in tables 2, 3, and 4.

In adversarial setups, a specific weight (γ) (see algorithm 1), is assigned to the adversary, balancing model accuracy with bias reduction. For private learning scenarios, a privacy budget (ϵ) per dataset was chosen with a cap below 15.

Table 1: Privacy parameters per dataset

Dataset	$(\epsilon - \delta)$	Noise Multiple	Batch Size	Adversary Weight
Adult	$(4, 10^{-6})$	0.5	16	0.2
COMPAS	$(7.7, 10^{-6})$	0.5	4	0.5
hate Speech	$(13, 10^{-6})$	0.5	64	0.1

Although this is an arbitrary cap, our choice was informed by considering reports that companies such as Apple use much larger privacy budgets in practice Tang et al. (2017). In the present work, we only considered one privacy budget per dataset. The privacy budgets applied for each dataset are determined by adjusting mini-batch sizes and noise multipliers.

4.5 EVALUATION METRICS

Performance Assessment: To address class imbalances in the datasets, we evaluate our models using both micro ($F1-w$) and macro F1 scores ($F1-m$), following the performance evaluation of Kang et al. (2022).

Fairness Assessment: Equality of odds is often used in the context of fairness-aware machine learning and algorithmic fairness Dwork et al. (2012); Hardt et al. (2016); Zafar et al. (2017). By evaluating and enforcing Equality of Odds, we can assess whether a model is treating different groups fairly. Significant differences between TPR and FPR across groups indicate that the model may be unfairly favoring or penalizing certain groups. The Equality of Odds metric ensures consistent True Positive Rates (TPR) and False Positive Rates (FPR). Assuming two groups, as A and B , Equality of Odds can be enforced as follows:

$$\Delta TPR = |TPR_A - TPR_B|, \quad \Delta FPR = |FPR_A - FPR_B|.$$

$$EoO = \sqrt{(\Delta TPR)^2 + (\Delta FPR)^2} \tag{3}$$

5 RESULTS

All evaluation metrics are computed using test sets, which constitute 20% of the entire datasets. For each training session, these test instances are randomly selected anew. Our results show that while standalone DP models generally performed worse, adversarial debiasing performed robustly, often challenging or exceeding baseline results.

Table 2: Evaluation metrics on Adult Dataset

Method	Micro F1 Score	Macro F1 Score	Attribute	Equal Odds Difference
Baseline	0.80 (± 0.002)	0.68 (± 0.005)	Race	-0.10 (± 0.02)
			Sex	-0.28 (± 0.01)
DP	0.74 (± 0.03)	0.65 (± 0.03)	Race	-0.10 (± 0.10)
			Sex	-0.19 (± 0.15)
Debias	0.79 (± 0.01)	0.66 (± 0.02)	Race	-0.02 (± 0.07)
			Sex	-0.008 (± 0.17)
DP + Debias	0.79 (± 0.004)	0.67 (± 0.03)	Race	-0.04 (± 0.05)
			Sex	-0.09 (± 0.05)

Table 3: Evaluation metrics on COMPAS Dataset

Method	Micro F1 Score	Macro F1 Score	Attribute	Equal Odds Difference
Baseline	0.66 (± 0.01)	0.65 (± 0.02)	Race	-0.12 (± 0.04)
			Sex	-0.11 (± 0.05)
DP	0.62 (± 0.02)	0.58 (± 0.05)	Race	-0.09 (± 0.12)
			Sex	-0.13 (± 0.08)
Debias	0.65 (± 0.01)	0.65 (± 0.01)	Race	-0.06 (± 0.1)
			Sex	-0.03 (± 0.08)
DP + Debias	0.64 (± 0.01)	0.63 (± 0.02)	Race	-0.08 (± 0.06)
			Sex	-0.04 (± 0.06)

Table 4: Evaluation metrics on Hate Speech Dataset

Method	Micro F1 Score	Macro F1 Score	Attribute	Equal Odds Difference
Baseline	0.78 (± 0.01)	0.75 (± 0.01)	Gender	-0.01 (± 0.01)
			Ethnicity	0.08 (± 0.01)
			Age	-0.08 (± 0.01)
			Country	-0.04 (± 0.01)
DP	0.76 (± 0.007)	0.68 (± 0.02)	Gender	0.02 (± 0.01)
			Ethnicity	0.07 (± 0.01)
			Age	-0.14 (± 0.01)
			Country	0.04 (± 0.01)
Debias	0.82 (± 0.005)	0.79 (± 0.003)	Gender	0.006 (± 0.008)
			Ethnicity	0.06 (± 0.005)
			Age	-0.08 (± 0.01)
			Country	0.04 (± 0.008)
DP + Debias	0.70 (± 0.02)	0.51 (± 0.05)	Gender	0.003 (± 0.006)
			Ethnicity	-0.036 (± 0.01)
			Age	-0.040 (± 0.03)
			Country	0.01 (± 0.006)

A possible explanation for the better performance of adversarial debiasing is that the technique emphasizes gradient propagation that resists favoring adversaries. With multiple sensitive attributes, the adversarial process’s complexity heightens, affecting gradient dynamics and necessitating nuanced tuning of hyper-parameters like α to avoid unintended biases.

In situations involving both multiple attributes and DP, performance generally takes a large hit. Noise introduced via differentially private optimization can disrupt adversarial debiasing’s balance. This noise can confuse the adversary in discerning genuine data patterns and biases, demanding gradient re-calibration for optimal fairness. Results for the hate speech dataset show that handling multiple sensitive attributes coupled with differential privacy introduces considerable complexity, which leads to large differences in performance. The *Debias-only* model performed best here.

6 CONCLUSION

Our study across four datasets shows that adversarial debiasing can remove multi-attribute bias, evident in both private and non-private contexts. This presents a tangible solution to the prominent challenge of ensuring fairness and privacy in machine learning implementations. This research shows the potential of concurrently using differential privacy and adversarial training in promoting privacy and fairness within machine learning models, but also that there are limits to upholding model performance under strict privacy conditions.

In summary, the approach provides the following advantages: reasons:

- **End-to-End Framework:** Adversarial debiasing offers an integrated solution, eliminating the need for multiple stages of interventions, which can complicate fairness and privacy integration.
- **Modularity of Differential Privacy:** Differential privacy is seamlessly integrated during training via a differentially private optimizer. This ensures both unbiased learning and individual data protection.
- **Handles Multiple Sensitive Attributes:** The method can debias several attributes at once, addressing intersectional biases like race, gender, and age, making it ideal for real-world applications.
- **Allows Multiple Fairness Definitions:** The adversary module is adaptable to various fairness metrics, such as demographic parity or equal opportunity. It’s versatile for both continuous and discrete outputs or protected variables.
- **Model-Agnostic:** Suitable for diverse data types like text or images, adversarial debiasing’s model-agnostic approach only requires prior knowledge of the bias variable for effective debiasing.

6.1 NOTES FOR FURTHER RESEARCH

Future research in adversarial debiasing and machine learning fairness should address several key areas. First, it is essential to get a better understanding of the complex effects of debiasing multiple attributes on performance. Second, generalizability of the approach should be tested by involving larger and more diverse datasets. Third, a shift from analyses confined to single privacy budgets towards a broader evaluation across varied privacy scenarios can discern performance patterns and trade-offs between privacy and efficiency. Fourth, comparing adversarial debiasing with other fairness methodologies can highlight unique strengths and potential drawbacks. Fifth, the sensitivity of adversarial debiasing methods to hyper-parameters deserves attention, both to gauge stability and to establish best practices. Finally, the potential of pre-training both the predictor and adversary networks, is worth exploring further, given its positive implications in certain learning contexts ??.

6.2 REPRODUCIBILITY

In efforts to facilitate reproducibility of our work, we provide our code as supplementary material to this work. The code includes all data processing steps taken for each dataset, and ensures network parameters are initialized with the same seed for random number generation.

REFERENCES

- Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.
- Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International conference on machine learning*, pp. 60–69. PMLR, 2018.
- Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias. 2016.
- Rachel K. E. Bellamy, Kuntal Dey, Michael Hind, Samuel C. Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. AI Fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, October 2018. URL <https://arxiv.org/abs/1810.01943>.
- Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5:135–146, 2017.
- Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security 19)*, pp. 267–284, 2019.
- Simon Caton and Christian Haas. Fairness in machine learning: A survey. *ACM Computing Surveys*, 2020.
- Kenneth Ward Church. Word2vec. *Natural Language Engineering*, 23(1):155–162, 2017.
- Jan Deriu, Aurelien Lucchi, Valeria De Luca, Aliaksei Severyn, Simon Müller, Mark Cieliebak, Thomas Hofmann, and Martin Jaggi. Leveraging large amounts of weakly supervised data for multi-language sentiment classification. In *Proceedings of the 26th international conference on world wide web*, pp. 1045–1052, 2017.
- Cynthia Dwork. Differential privacy. In *International colloquium on automata, languages, and programming*, pp. 1–12. Springer, 2006.
- Cynthia Dwork. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pp. 1–19. Springer, 2008.
- Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pp. 214–226, 2012.
- Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014.
- Andrew Frank. Uci machine learning repository. <http://archive.ics.uci.edu/ml>, 2010.
- Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*, pp. 619–633, 2018.
- Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.
- Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. *Advances in neural information processing systems*, 29, 2016.

- Hongsheng Hu, Zoran Salcic, Lichao Sun, Gillian Dobbie, Philip S Yu, and Xuyun Zhang. Membership inference attacks on machine learning: A survey. *ACM Computing Surveys (CSUR)*, 54(11s):1–37, 2022.
- Xiaolei Huang, Linzi Xing, Franck Deroncourt, and Michael J Paul. Multilingual twitter corpus and baselines for evaluating demographic bias in hate speech recognition. *arXiv preprint arXiv:2002.10361*, 2020.
- Matthew Jagielski, Michael Kearns, Jieming Mao, Alina Oprea, Aaron Roth, Saeed Sharifi-Malvajerdi, and Jonathan Ullman. Differentially private fair learning. In *International Conference on Machine Learning*, pp. 3000–3008. PMLR, 2019.
- Zhanglong Ji, Zachary C Lipton, and Charles Elkan. Differential privacy and machine learning: a survey and review. *arXiv preprint arXiv:1412.7584*, 2014.
- Jian Kang, Tiankai Xie, Xintao Wu, Ross Maciejewski, and Hanghang Tong. Infofair: Information-theoretic intersectional fairness. In *2022 IEEE International Conference on Big Data (Big Data)*, pp. 1455–1464. IEEE, 2022.
- Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. Preventing fairness gerrymandering: Auditing and learning for subgroup fairness. In *International conference on machine learning*, pp. 2564–2572. PMLR, 2018.
- Ron Kohavi. Scaling up the accuracy of naive-bayes classifiers: a. In *Proceedings of the Second International Conference on*, 1996.
- Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS’07)*, pp. 94–103. IEEE, 2007.
- Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54(6):1–35, 2021.
- Mathias PM Parisot, Balazs Pejo, and Dayana Spagnuolo. Property inference attacks on convolutional neural networks: Influence and implications of target model’s complexity. *arXiv preprint arXiv:2104.13061*, 2021.
- Dana Pessach and Erez Shmueli. A review on fairness in machine learning. *ACM Computing Surveys (CSUR)*, 55(3):1–44, 2022.
- Samantha Petti and Abraham Flaxman. Differential privacy in the 2020 us census: what will it do? quantifying the accuracy/privacy tradeoff. *Gates open research*, 3, 2019.
- Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, 2017.
- Jun Tang, Aleksandra Korolova, Xiaolong Bai, Xueqiang Wang, and Xiaofeng Wang. Privacy loss in apple’s implementation of differential privacy on macos 10.12. *arXiv preprint arXiv:1709.02753*, 2017.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. In *Artificial intelligence and statistics*, pp. 962–970. PMLR, 2017.
- Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pp. 335–340, 2018.
- Jingwen Zhao, Yunfang Chen, and Wei Zhang. Differential privacy preservation in deep learning: Challenges, opportunities and solutions. *IEEE Access*, 7:48901–48911, 2019.