

Erasure of Unaligned Attributes from Neural Representations

Shun Shao* Yftah Ziser* Shay B. Cohen

Institute for Language, Cognition and Computation

School of Informatics, University of Edinburgh

10 Crichton Street, Edinburgh, EH8 9AB, UK

s.shao-11@inf.ed.ac.uk yftah.ziser@inf.ed.ac.uk

scohen@inf.ed.ac.uk

Abstract

We present the Assignment-Maximization Spectral Attribute removal (AMSAL) algorithm, which erases information from neural representations when the information to be erased is implicit rather than directly being aligned to each input example. Our algorithm works by alternating between two steps. In one, it finds an assignment of the input representations to the information to be erased, and in the other, it creates projections of both the input representations and the information to be erased into a joint latent space. We test our algorithm on an extensive array of datasets, including a Twitter dataset with multiple guarded attributes, the BiasBios dataset, and the BiasBench benchmark. The latter benchmark includes four datasets with various types of protected attributes. Our results demonstrate that bias can often be removed in our setup. We also discuss the limitations of our approach when there is a strong entanglement between the main task and the information to be erased.¹

1 Introduction

Developing a methodology for adjusting neural representations to preserve user privacy and avoid encoding bias in them has been an active area of research in recent years. Previous work shows it is possible to erase undesired information from representations so that downstream classifiers cannot use that information in their decision-making process. This previous work assumes that this sensitive information (or **guarded** attributes, such as gender or race) is available for each input instance. These guarded attributes, however, are sensitive, and obtaining them on a large scale is often challenging and, in some cases, not feasible (Han et al., 2021b). For example, Blodgett et al. (2016)

studied the characteristics of African-American English on Twitter, and could not couple the ethnicity attribute directly with the tweets they collected due to the attribute’s sensitivity.

This paper introduces a novel debiasing setting in which the guarded attributes are not paired up with each input instance and an algorithm to remove information from representations in that setting. In our setting, we assume that each neural input representation is coupled with a guarded attribute value, but this assignment is unavailable. In cases where the domain of the guarded attribute is small (for example, with binary attributes), this means that the guarded attribute information consists of **priors** with respect to the whole population and not instance-level information.

The intuition behind our algorithm is that if we were to find a strong correlation between the input variable and a set of guarded grounded attributes either in the form of an unordered list of **records** or as priors, then it is unlikely to be coincidental if the sample size is sufficiently large (§3.5). We implement this intuition by jointly finding projections of the input samples and the guarded attributes into a joint embedding space *and* an alignment between the two sets in that joint space.

Our resulting algorithm (§3), the Alignment-Maximization Spectral Attribute removal algorithm (AMSAL), is a coordinate-ascent algorithm reminiscent of the hard expectation-maximization algorithm (hard EM; MacKay, 2003). It first loops between two Alignment and Maximization steps, during which it finds an alignment (A) based on existing projections and then projects the representations and guarded attributes into a joint space based on an existing alignment (M). After these two steps are iteratively repeated and an alignment is identified, the algorithm takes another step to erase information from the input representations based on the projections identified. This step closely follows the work of Shao et al.

*Equal contribution.

¹Our code is available at <https://github.com/jasonshaoshun/AMSAL>.

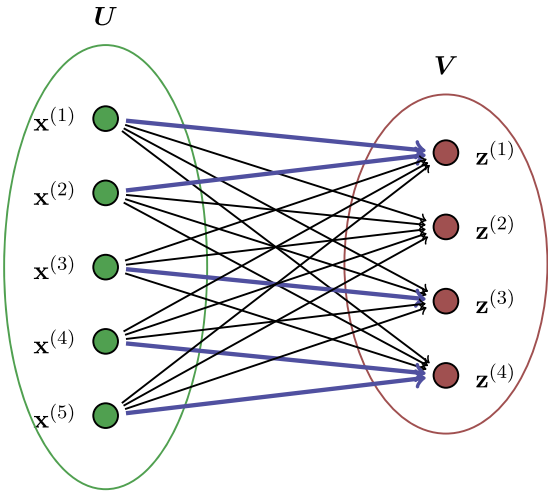


Figure 1: A depiction of the problem setting and solution. The inputs are aligned to each guarded sample, based on strength using two projections U and V . We solve a bipartite matching problem to find the blue edges, and then recalculate U and V .

(2023), who use Singular Value Decomposition to remove principal directions of the covariance matrix between the input examples and the guarded attributes. Figure 1 depicts a sketch of our setting and the corresponding algorithm, with \mathbf{x}_i being the input representations and \mathbf{z}_j being the guarded attributes. Our algorithm is modular: While our use of the algorithm of Shao et al. (2023) for the removal step is natural due to the nature of the AM steps, a user can use any such algorithm to erase the information from the input representations (§3.4).

Our contributions are as follows: (1) We propose a new setup for removing guarded information from neural representations where there are few or no labeled guarded attributes; (2) We present a novel two-stage coordinate-ascent algorithm that iteratively improves (a) an alignment between guarded attributes and neural representations; and (b) information removal projections.

Using an array of datasets, we perform extensive experiments to assess how challenging our setup is and whether our algorithm is able to remove information without having aligned guarded attributes (§4). We find in several cases that little information is needed to align between neural representations and their corresponding guarded attributes. The consequence is that it is possible to erase the information such guarded attributes provide from the neural representations while

preserving the information needed for the main task decision-making. We also study the limitations of our algorithm by experimenting with a setup where it is hard to distinguish between the guarded attributes and the downstream task labels when aligning the neural representations with the guarded attributes (§4.5).

2 Problem Formulation and Notation

For an integer n we denote by $[n]$ the set $\{1, \dots, n\}$. For a vector \mathbf{v} , we denote by $\|\mathbf{v}\|_2$ its ℓ_2 norm. For two vectors \mathbf{v} and \mathbf{u} , by default in column form, $\langle \mathbf{v}, \mathbf{u} \rangle = \mathbf{v}^\top \mathbf{u}$ (dot product). Matrices and vectors are in boldface font (with uppercase or lowercase letters, respectively). Random variable vectors are also denoted by boldface uppercase letters. For a matrix \mathbf{A} , we denote by a_{ij} the value of cell (i, j) . The Frobenius norm of a matrix \mathbf{A} is $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} a_{ij}^2}$. The spectral norm of a matrix is $\|\mathbf{A}\|_2 = \max_{\|\mathbf{x}\|_2=1} \|\mathbf{A}\mathbf{x}\|_2$. The expectation of a random variable \mathbf{T} is denoted by $\mathbb{E}[\mathbf{T}]$.

In our problem formulation, we assume three random variables: $\mathbf{X} \in \mathbb{R}^d$, $\mathbf{Y} \in \mathbb{R}$, and $\mathbf{Z} \in \mathbb{R}^{d'}$, such that $d' \leq d$ and the expectation of all three variables is 0 (see Shao et al., 2023). Samples of \mathbf{X} are the inputs for a classifier to predict corresponding samples of \mathbf{Y} . The random vector \mathbf{Z} represents the guarded attributes. We want to maintain the ability to predict \mathbf{Y} from \mathbf{X} , while minimizing the ability to predict \mathbf{Z} from \mathbf{X} .

We assume n samples of (\mathbf{X}, \mathbf{Y}) and m samples of \mathbf{Z} , denoted by $(\mathbf{x}^{(i)}, \mathbf{y}^{(i)})$ for $i \in [n]$, and $\mathbf{z}^{(i)}$ for $i \in [m]$ ($m \leq n$). While originally, these samples were generated jointly from the underlying distribution $p(\mathbf{X}, \mathbf{Y}, \mathbf{Z})$, we assume a shuffling of the \mathbf{Z} samples in such a way that we are only left with m samples that are unique (no repetitions) and an underlying unknown many-to-one mapping $\pi : [n] \rightarrow [m]$ that maps each $\mathbf{x}^{(i)}$ to its original $\mathbf{z}^{(j)}$.

The problem formulation is such that we need to remove the information from the x s in such a way that we consider the samples of z s as a set. In our case, we do so by iterating between trying to infer π , and then using standard techniques to remove the information from x s based on their alignment to the corresponding z s.

Singular Value Decomposition Let $\mathbf{A} = \mathbb{E}[\mathbf{X}\mathbf{Z}^\top]$, the matrix of cross-covariance between

\mathbf{X} and \mathbf{Z} . This means that $\mathbf{A}_{ij} = \text{Cov}(X_i, Z_j)$ for $i \in [d]$ and $j \in [d']$.

For any two vectors, $\mathbf{a} \in \mathbb{R}^d, \mathbf{b} \in \mathbb{R}^{d'}$, the following holds due to the linearity of expectation:

$$\mathbf{a}\mathbf{b}^\top = \text{Cov}(\mathbf{a}^\top \mathbf{X}, \mathbf{b}^\top \mathbf{Z}). \quad (1)$$

Singular value decomposition on \mathbf{A} , in this case, finds the ‘‘principal directions’’: directions in which the projection of \mathbf{X} and \mathbf{Z} maximize their covariance. The projections are represented as two matrices $\mathbf{U} \in \mathbb{R}^{d \times d}$ and $\mathbf{V} \in \mathbb{R}^{d' \times d'}$. Each column in these matrices plays the role of the vectors \mathbf{a} and \mathbf{b} in Eq. 1. SVD finds \mathbf{U} and \mathbf{V} such that for any $i \in [d']$ it holds that:

$$\text{Cov}(\mathbf{U}_i^\top \mathbf{X}, \mathbf{V}_i^\top \mathbf{Z}) = \max_{(\mathbf{a}, \mathbf{b}) \in \mathcal{O}_i} \text{Cov}(\mathbf{a}^\top \mathbf{X}, \mathbf{b}^\top \mathbf{Z}),$$

where \mathcal{O}_i is the set of pairs of vectors (\mathbf{a}, \mathbf{b}) such that $\|\mathbf{a}\|_2 = \|\mathbf{b}\|_2 = 1$, \mathbf{a} is orthogonal to $\mathbf{U}_1, \dots, \mathbf{U}_{i-1}$ and similarly, \mathbf{b} is orthogonal to $\mathbf{V}_1, \dots, \mathbf{V}_{i-1}$.

Shao et al. (2023) showed that SVD in this form can be used to debias representations. We calculate SVD between \mathbf{X} and \mathbf{Z} and then prune out the principal directions that denote the highest covariance. We will use their method, SAL (Spectral Attribute removalL), in the rest of the paper. See also §3.4.

3 Methodology

We view the problem of information removal with unaligned samples as a joint optimization problem of: (a) finding the alignment; (b) finding the projection that maximizes the covariance between the alignments, and using its complement to project the inputs. Such an optimization, in principle, is intractable, so we break it down into two coordinate-ascent style steps: A-step (in which the alignment is identified as a bipartite graph matching problem) and M-step (in which based on the previously identified alignment, a maximal-covariance projection is calculated). Formally, the maximization problem we solve is:

$$(\mathbf{U}, \mathbf{V}, \pi) = \arg \max_{\mathbf{U}, \mathbf{V}, \pi} \sum_{i=1}^n (\mathbf{x}^{(i)})^\top \mathbf{U} \mathbf{V}^\top \mathbf{z}^{(i)}, \quad (2)$$

where we constrain \mathbf{U} and \mathbf{V} to be matrices with orthonormal columns in $\mathbb{R}^{n \times k}$.

Inputs: Samples $\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}$, and $\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}$.

Algorithm: (calculate projection which removes information from the x s that is in the z s)

Initialize π randomly to a function from $[n]$ to $[m]$.

Repeat the following for T iterations:

- (M-step) Using π , let Ω_π as in Eq. 3. Calculate SVD on Ω_π to calculate $(\mathbf{U}, \Sigma, \mathbf{V})$.
- (A-step) With \mathbf{U} and \mathbf{V} as above, with top k singular vectors, find π by solving the problem as in Eq. 2.

Return: The singular vectors from \mathbf{U} that have lowest singular values.

Figure 2: The main Assignment-Maximization Spectral Attribute removalL (AMSAL) algorithm for removal of information without alignment between samples of \mathbf{X} and \mathbf{Z} .

Note that the sum in the above equation has a term per pair of $(\mathbf{x}^{(i)}, \mathbf{z}^{\pi(i)})$, which enables us to frame the A-step as an integer linear programming (ILP) problem (§3.1). The full algorithm is given in Figure 2, and we proceed in the next two steps to further explain the A-step and the M-step.

3.1 A-step (Guarded Sample Assignment)

In the Assignment Step, we are required to find a many-to-one alignment $\pi : [n] \rightarrow [m]$ between $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(n)}\}$ and $\{\mathbf{z}^{(1)}, \dots, \mathbf{z}^{(m)}\}$. Given \mathbf{U} and \mathbf{V} from the previous M-step, we can find such an assignment by solving the following optimization problem:

$$\arg \max_{\pi} \sum_{i=1}^n \langle \mathbf{U}^\top \mathbf{x}^{(i)}, \mathbf{V}^\top \mathbf{z}^{(\pi(i))} \rangle.$$

This maximization problem can be formulated as an integer linear program of the following form:

$$\begin{aligned} \max_{\mathbf{P} \in \{0,1\}^{n \times m}} \quad & \sum_{j=1}^m \sum_{i=1}^n p_{ij} \langle \mathbf{U}^\top \mathbf{x}^{(i)}, \mathbf{V}^\top \mathbf{z}^{(j)} \rangle \\ \text{s.t. } \forall i. \quad & \sum_{j=1}^m p_{ij} = 1, \\ \forall j. \quad & b_{0j} \leq \sum_{i=1}^n p_{ij} \leq b_{1j}. \end{aligned} \quad (3)$$

This is a solution to an assignment problem (Kuhn, 1955; Ramshaw and Tarjan, 2012), where p_{ij} denotes whether $\mathbf{x}^{(i)}$ is associated with the (type of) guarded attribute $\mathbf{z}^{(j)}$. The values (b_{0j}, b_{1j}) determine lower and upper bounds on the number of x s a given $\mathbf{z}^{(j)}$ can be assigned to. While a standard assignment problem can be solved efficiently using the Hungarian method of Kuhn (1955), we choose to use the ILP formulation, as it enables us to have more freedom in adding constraints to the problem, such as the lower and upper bounds.

3.2 M-step (Covariance Maximization)

The result of an A-step is an assignment π such that $\pi(i) = j$ implies $\mathbf{x}^{(i)}$ was deemed as aligned to \mathbf{z}_j . With that π in mind, we define the following empirical covariance matrix $\Omega_\pi \in \mathbb{R}^{d \times d'}$:

$$\Omega_\pi = \sum_{i=1}^n \mathbf{x}^{(i)} \left(\mathbf{z}^{(\pi(i))} \right)^\top. \quad (4)$$

We then apply SVD on Ω_π to get new U and V that are used in the next iteration of the algorithm with the A-step, if the algorithm continues to run. When the maximal number of iterations is reached, we follow the work of Shao et al. (2023) in using a truncated part of U to remove the information from the x s. We do that by projecting $\mathbf{x}^{(i)}$ using the singular vectors of U with the smallest singular values. These projected vectors co-vary the least with the guarded attributes, assuming the assignment in the last A-step was precise. This method has been shown by Shao et al. (2023) to be highly effective and efficient in debiasing neural representations.

3.3 A Matrix Formulation of the AM Steps

Let $\mathbf{e}_1, \dots, \mathbf{e}_m$ be the standard basis vectors. This means \mathbf{e}_i is a vector of length m with 0 in all coordinates except for the i th coordinate, where it is 1.

Let \mathcal{E} be the set of all matrices E where each $E \in \mathcal{E}$ is such that $E \in \mathbb{R}^{n \times m}$ and each row is one of \mathbf{e}_i , $i \in [m]$. In that case, $E\mathbf{Z}^\top$ is an $n \times d'$ matrix, such that the j th row is a copy of the i th column of $\mathbf{Z} \in \mathbb{R}^{d' \times n}$. Therefore, the AM steps can be viewed as solving the following maximization problem using coordinate ascent:

$$\arg \max_{E \in \mathcal{E}, U, V, \Sigma} \|U^\top \Sigma V - E\mathbf{Z}^\top\|_F^2,$$

where U, V are orthonormal matrices, and Σ is a diagonal matrix with non-negative elements. This corresponds to the SVD of the matrix $\mathbf{X}\mathbf{E}\mathbf{Z}^\top$.

In that case, the matrix E can be directly mapped to an assignment in the form of π , where $\pi(i)$ would be the j such that the j th coordinate in the i th row of E is non-zero.

3.4 Removal Algorithm

The AM steps are best suited for the removal of information through SVD with an algorithm such as SAL. This is because AM steps are optimizing an objective of the same type of SAL—relying on the projections U and V to project the inputs and guarded representations into a joint space. However, a by-product of the algorithm in Figure 2 is an assignment function π that aligns between the inputs and the guarded representations.

With that assignment, other removal algorithms can be used, for example, the algorithm of Ravfogel et al. (2020). We experiment with this idea in §4.

3.5 Justification of the AM Steps

We next provide a justification of our algorithm (which may be skipped on a first reading). Our justification is based on the observation that if indeed \mathbf{X} and \mathbf{Z} are linked together (this connection is formalized as a latent variable in their joint distribution), then for a given sample that is permuted, the singular values of Ω will be larger the closer the permutation is to the identity permutation. This justifies finding such a permutation that maximizes the singular values in an SVD of Ω .

More Details Let $\iota : [n] \rightarrow [n]$ be the identity permutation, $\iota(i) = i$. We will assume the case in which $n = m$ (but the justification can be generalized to the case $m < n$), and that the underlying joint distribution $p(\mathbf{X}, \mathbf{Z})$ is mediated by a latent variable \mathbf{H} , such that

$$p(\mathbf{X}, \mathbf{Z}, \mathbf{H}) = p(\mathbf{H})p(\mathbf{X} | \mathbf{H})p(\mathbf{Z} | \mathbf{H}). \quad (5)$$

This implies there is a latent variable that connects \mathbf{X} and \mathbf{Z} , and that the joint distribution $p(\mathbf{X}, \mathbf{Z})$ is a mixture through \mathbf{H} .

Proposition 1 (informal). *Let $\{(\mathbf{x}^{(i)}, \mathbf{z}^{(i)})\}$ be a sample of size n from the distribution in Eq. 5. Let π be a permutation over $[n]$ uniformly sampled from the set of permutations. Then with high likelihood, the sum of the singular values of Ω_π is smaller than the sum of singular values under Ω_v .*

For full details of this claim, see Appendix A.

4 Experiments

In our experiments, we test several combinations of algorithms. We use the k -means (KMEANS) as a substitute for the AM steps as a baseline for the assignment step of x s to z s. In addition, for the removal step (once an assignment has been identified), we test two algorithms: SAL (Shao et al., 2023; resulting in AMSAL) and INLP (Ravfogel et al., 2020). We also compare these two algorithms in *oracle* mode (in which the assignment of guarded attributes to inputs is known), to see the loss in performance that happens due to noisy assignments from the AM or k -means algorithm (ORACLE-SAL and ORACLE-INLP).

When running the AM algorithm or k -means, we execute it with three random seeds (see also §4.6) for a maximum of a hundred iterations and choose the projection matrix with the largest objective value over all seeds and iterations. For the slack variables (b_{0j} and b_{1j} variables in Eq. 3), we use 20%–30% above and below the baseline of the guarded attribute priors according to the training set. With the SAL methods, we remove the number of directions according to the rank of the Ω matrix (between 2 to 6 in all experiments overall).

In addition, we experiment with a partially supervised assignment process, in which a small seed dataset of aligned x s and z s is provided to the AM steps. We use it for model selection: Rather than choosing the assignment with the highest SVD objective value, we choose the assignment with the highest accuracy on this seed dataset. We refer to this setting as PARTIAL (for “partially supervised assignment”).

Finally, in the case of a gender-protected attribute, we compare our results against a baseline in which the input \mathbf{x} is compared against a list of words stereotypically associated with the genders of male or female.² Based on the overlap with

these two lists, we heuristically assign the gender label to \mathbf{x} and then run SAL or INLP (rather than using the AM algorithm). While this word-list heuristic is plausible in the case of gender, it is not as easy to derive in the case of other protected attributes, such as age or race. We give the results for this baseline using the marker WL in the corresponding tables.

Main Findings Our overall main finding shows that our novel setting in which guarded information is erased from individually unaligned representations is viable. We discovered that AM methods perform particularly well when dealing with more complex bias removal scenarios, such as when multiple guarded attributes are present. We also found that having similar priors for the guarded attributes and downstream task labels may lead to poor performance on the task at hand. In these cases, using a small amount of supervision often effectively helps reduce bias while maintaining the utility of the representations for the main classification of the regression problem. Finally, our analysis of alignment stability shows that our AM algorithm often converges to suitable solutions that align \mathbf{X} with \mathbf{Z} .

Due to the unsupervised nature of our problem setting, we advise validating the utility of our method in the following way. Once we run the AM algorithm, we check whether there is a high-accuracy alignment between \mathbf{X} and \mathbf{Y} (rather than \mathbf{Z} , which is unavailable). If this alignment is accurate, then we run the risk of significantly damaging task performance. An example is given in §4.5.

4.1 Word Embedding Debiasing

As a preliminary assessment of our setup and algorithms, we apply our methods to GloVe word embeddings to remove gender bias, and follow the previous experiment settings of this problem (Bolukbasi et al., 2016; Ravfogel et al., 2020; Shao et al., 2023). We considered only the 150,000 most common words to ensure the embedding quality and omitted the rest. We sort the remaining embeddings by their projection on the he-she direction. Then we consider the top 7,500 word embeddings as male-associated words ($z = 1$) and the bottom 7,500 as female-associated words ($z = -1$).

Our findings are that both the k -means and the AM algorithms perfectly identify the alignment

²<https://tinyurl.com/33bzddtw>.

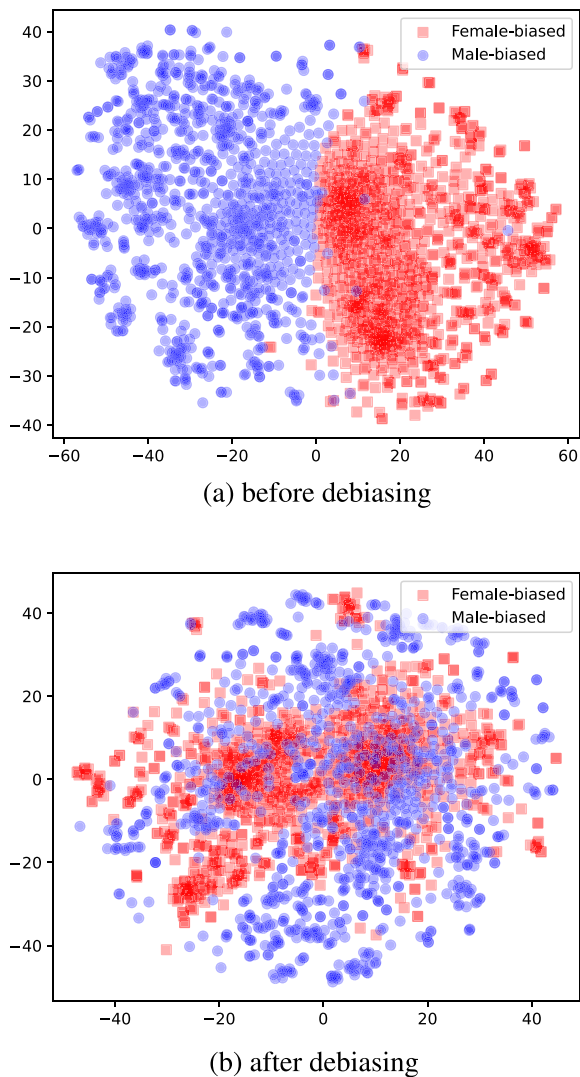


Figure 3: A t-SNE visualization of the word embeddings before and after gender information removal. In (a) we see the embeddings naturally cluster into the corresponding gender.

between the word embeddings and their associated gender label (100%). Indeed, the dataset construction itself follows a natural perfect clustering that these algorithms easily discover. Since the alignments are perfectly identified, the results of predicting the gender from the word embeddings after removal are identical to the oracle case. These results are quite close to the results of a random guess, and we refer the reader to Shao et al. (2023) for details on experiments with SAL and INLP for this dataset. Considering Figure 3, it is evident that our algorithm essentially follows a natural clustering of the word embeddings into two clusters, female and male, as the embeddings are highly separable in this case. This is why the alignment score of \mathbf{X} (embedding) to \mathbf{Z} (gender)

is perfect in this case. This finding indicates that this standard word embedding dataset used for debiasing is *trivial to debias*—debiasing can be done even without knowing the identity of the stereotypical gender associated with each word.

4.2 BiasBios Results

De-Arteaga et al. (2019) presented the BiasBios dataset, which consists of self-provided biographies paired with the profession and gender of their authors. A list of pronouns and names is used to obtain the authors’ gender automatically. They aim to expose the caveats of automated hiring systems by showing that even the simple task of predicting a candidate’s profession can be affected by the candidate’s gender, which is encoded in the biography representation. For example, we want to avoid one being identified as “he” or “she” in their biography, affecting the likelihood of them being classified as engineers or teachers.

We follow the setup of De-Arteaga et al. (2019), predicting a candidate’s professions (\mathbf{y}), based on a self-provided short biography (\mathbf{x}), aiming to remove any information about the candidate’s gender (\mathbf{z}). Due to computational constraints, we use only random 30K examples to learn the projections with both SAL and INLP (whether in the unaligned or aligned setting). For the classification problem, we use the full dataset. To obtain vector representations for the biographies, we use two different encoders, FastText word embeddings (Joulin et al., 2016), and BERT (Devlin et al., 2019). We stack a multi-class classifier on top of these representations, as there are 28 different professions. We use 20% of the training examples for the PARTIAL setting. For BERT, we followed De-Arteaga et al. (2019) in using the last CLS token state as the representation of the whole biography. We used the BERT model `bert-base-uncased`.

Evaluation Measures We use an extension of the True Positive Rate (TPR) gap, the root mean square (RMS) TPR gap of all classes, for evaluating bias in a multiclass setting. This metric was suggested by De-Arteaga et al. (2019), who demonstrated it is significantly correlated with gender imbalances, which often lead to unfair classification. The higher the metric value is, the bigger the gap between the two categories (for example, between male and female) for the

Model	Task Acc.	TPR-GAP
BERTMODEL	0.79	0.20
+ AMINLP	↓0.12 0.67	↓0.12 0.09
+ KMEANS + INLP	↓0.11 0.68	↓0.12 0.08
+ ORACLEINLP	↓0.11 0.68	↓0.12 0.08
+ PARTIALINLP	↓0.12 0.67	↓0.13 0.08
+ AMSAL	0.79	↓0.02 0.18
+ KMEANS + SAL	0.79	↓0.02 0.18
+ ORACLESAL	0.79	↓0.02 0.18
+ PARTIALSAL	0.79	↓0.02 0.18
+ WL + SAL	0.79	↓0.02 0.18
+ WL + INLP	↓0.12 0.68	↓0.12 0.08
FASTTEXT	0.77	0.20
+ AMINLP	↓0.05 0.73	↑0.01 0.21
+ KMEANS + INLP	↓0.08 0.69	0.19
+ ORACLEINLP	↓0.03 0.74	↓0.10 0.09
+ PARTIALINLP	↓0.04 0.74	↓0.04 0.16
+ AMSAL	↓0.03 0.74	↓0.03 0.17
+ KMEANS + SAL	↓0.04 0.73	↓0.02 0.17
+ ORACLESAL	↓0.01 0.76	↓0.08 0.12
+ PARTIALSAL	↓0.01 0.76	↓0.02 0.18
+ WL + SAL	↓0.01 0.76	↓0.08 0.12
+ WL + INLP	↓0.03 0.74	↓0.10 0.10

Table 1: BiasBios dataset results. The top part uses BERT embeddings to encode the biographies, while the bottom part uses FastText embeddings.

specific main task prediction. For the profession classification, we report accuracy.

Results Table 1 provides the results for the biography dataset. We see that INLP significantly reduces the TPR-GAP in all settings, but this comes at a cost: The representations are significantly less useful for the main task of predicting the profession. When inspecting the alignments, we observe that their accuracy is quite high with BERT: 100% with k -means, 85% with the AM algorithm, and 99% with PARTIAL AM. For FastText, the results are lower, hovering around 55% for all three methods. The high BERT assignment performance indicates that the BiasBios BERT representations are naturally separated by gender. We also observe that the results of WL+SAL and WL+INLP are correspondingly identical to Oracle+SAL and Oracle+INLP. This comes as no surprise, as the gender label is derived from a similar word list, which enables the WL approach to get a nearly perfect alignment (over 96% agreement with the gender label).

4.3 BiasBench Results

Meade et al. (2022) followed an empirical study of an array of datasets in the context of debiasing. They analyzed different methods and tasks, and we follow their benchmark evaluation to assess our AMSAL algorithm and other methods in the context of our new setting. We include a short description of the datasets we use in this section. We include full results in Appendix B, with a description of other datasets. We also encourage the reader to refer to Meade et al. (2022) for details on this benchmark. We use 20% of the training examples for the PARTIAL setting.

StereoSet (Nadeem et al., 2021) This dataset presents a word completion test for a language model, where the completion can be stereotypical or non-stereotypical. The bias is then measured by calculating how often a model prefers the stereotypical completion over the non-stereotypical one. Nadeem et al. (2021) introduced the language model score to measure the language model usability, which is the percentage of examples for which a model prefers the stereotypical or non-stereotypical word over some unrelated word.

CrowS-Pairs (Nangia et al., 2020) This dataset includes pairs of sentences that are minimally different at the token level, but these differences lead to the sentence being either stereotypical or anti-stereotypical. The assessment measures how many times a language model prefers the stereotypical element in a pair over the anti-stereotypical element.

Results We start with an assessment of the BERT model for the CrowS-Pairs gender, race, and religion bias evaluation (Table 2). We observe that all approaches for gender, except AM+INLP, reduce the stereotype score. Race and religion are more difficult to debias in the case of BERT. INLP with k -means works best when no seed alignment data is provided at all, but when we consider PARTIALSAL, in which we use the alignment algorithm with some seed aligned data, we see that the results are the strongest. When we consider the RoBERTa model, the results are similar, with PARTIALSAL significantly reducing the bias. Our findings from Table 2 overall indicate that the ability to debias a representation *highly depends on the model that generates the representation*. In Table 10 we observe that the

Model	Stt. Score	Model	Stt. Score	Model	Stt. Score
Gender		Race		Religion	
BERT	57.25	BERT	62.33	BERT	62.86
+ AM + INLP	↑0.38 57.63	+ AM + INLP	↓1.75 60.58	+ AM + INLP	↑0.95 63.81
+ KMEANS + INLP	↓3.81 53.44	+ KMEANS + INLP	↑4.85 67.18	+ KMEANS + INLP	↑4.76 67.62
+ ORACLEINLP	↓4.58 52.67	+ ORACLEINLP	↑5.63 67.96	+ ORACLEINLP	↓1.91 60.95
+ PARTIALINLP	↓4.58 52.67	+ PARTIALINLP	↑5.63 67.96	+ PARTIALINLP	↓1.91 60.95
+ AMSAL	↓3.05 54.20	+ AMSAL	↑0.19 62.52	+ AMSAL	↑1.90 64.76
+ KMEANS + SAL	↓2.29 54.96	+ KMEANS + SAL	↑0.19 62.52	+ KMEANS + SAL	↑1.90 64.76
+ ORACLESAL	↓5.72 51.53	+ ORACLESAL	↑0.78 63.11	+ ORACLESAL	↑4.76 67.62
+ PARTIALSAL	↓5.72 51.53	+ PARTIALSAL	↑0.78 63.11	+ PARTIALSAL	↑4.76 67.62
ALBERT	48.09	ALBERT	62.52	ALBERT	60.00
+ AM + INLP	↑1.14 46.95	+ AM + INLP	↑0.98 36.50	+ AM + INLP	↓0.95 59.05
+ KMEANS + INLP	↑0.38 47.71	+ KMEANS + INLP	↑3.50 33.98	+ KMEANS + INLP	↑9.52 69.52
+ ORACLEINLP	↑4.58 43.51	+ ORACLEINLP	↓7.18 55.34	+ ORACLEINLP	↓2.86 57.14
+ PARTIALINLP	↑4.20 43.89	+ PARTIALINLP	↓7.18 55.34	+ PARTIALINLP	↓2.86 57.14
+ AMSAL	↑0.76 47.33	+ AMSAL	↓5.82 43.30	+ AMSAL	↑10.48 70.48
+ KMEANS + SAL	↓0.38 48.47	+ KMEANS + SAL	↓6.40 43.88	+ KMEANS + SAL	↑9.52 69.52
+ ORACLESAL	↑0.76 47.33	+ ORACLESAL	↓3.69 41.17	+ ORACLESAL	↑6.67 66.67
+ PARTIALSAL	↑0.76 47.33	+ PARTIALSAL	↓3.69 41.17	+ PARTIALSAL	↑6.67 66.67
RoBERTa	60.15	RoBERTa	63.57	RoBERTa	60.95
+ AM + INLP	↓3.45 56.70	+ AM + INLP	↓9.31 45.74	+ AM + INLP	↑4.76 65.71
+ KMEANS + INLP	↓7.66 52.49	+ KMEANS + INLP	↓6.79 43.22	+ KMEANS + INLP	↑2.86 63.81
+ ORACLEINLP	↓4.98 55.17	+ ORACLEINLP	↓1.75 61.82	+ ORACLEINLP	↑1.91 62.86
+ PARTIALINLP	↓4.98 55.17	+ PARTIALINLP	↓1.75 61.82	+ PARTIALINLP	↑1.91 62.86
+ AMSAL	↓3.45 56.70	+ AMSAL	↑1.74 65.31	+ AMSAL	↑12.38 73.33
+ KMEANS + SAL	↓3.83 56.32	+ KMEANS + SAL	↑1.74 65.31	+ KMEANS + SAL	↑12.38 73.33
+ ORACLESAL	↓8.81 48.66	+ ORACLESAL	↑3.48 67.05	+ ORACLESAL	↑10.48 71.43
+ PARTIALSAL	↓8.81 48.66	+ PARTIALSAL	↑3.48 67.05	+ PARTIALSAL	↑10.48 71.43
GPT-2	56.87	GPT-2	59.69	GPT-2	61.90
+ AM + INLP	↓6.11 50.76	+ AM + INLP	↓3.88 55.81	+ AM + INLP	61.90
+ KMEANS + INLP	↓2.67 54.20	+ KMEANS + INLP	↓1.16 58.53	+ KMEANS + INLP	61.90
+ ORACLEINLP	↓6.49 50.38	+ ORACLEINLP	↑0.19 59.88	+ ORACLEINLP	61.90
+ PARTIALINLP	↓6.11 50.76	+ PARTIALINLP	59.69	+ PARTIALINLP	61.90
+ AMSAL	↑1.15 58.02	+ AMSAL	↓4.65 55.04	+ AMSAL	↑3.81 65.71
+ KMEANS + SAL	↓3.05 53.82	+ KMEANS + SAL	↓5.43 45.74	+ KMEANS + SAL	↓1.90 60.00
+ ORACLESAL	56.87	+ ORACLESAL	↓4.85 54.84	+ ORACLESAL	↑15.24 77.14
+ PARTIALSAL	56.87	+ PARTIALSAL	↓4.85 54.84	+ PARTIALSAL	↑15.24 77.14

(a)

(b)

(c)

Table 2: (a) CrowS-Pairs Gender stereotype scores (Stt. score) in language models debiased by different debiasing techniques and assignment; (b) CrowS-Pairs Race stereotype scores; (c) CrowS-Pairs Religion stereotype scores. All models are deemed least biased if the stereotype score is 50%. The colored numbers are calculated as $||b - 50| - |s - 50||$ where b is the top row score and s is the corresponding system score.

representations, on average, are not damaged for most GLUE tasks. Additional analysis, included with a full version appendix, shows that the representations, on average, *are not damaged for most GLUE tasks*.

As Meade et al. (2022) have noted, when changing the representations of a language model to remove bias, we might cause such adjustments

that damage the usability of the language model. To test which methods possibly cause such an issue, we also assess the language model score on the StereoSet dataset in Table 3. We overall see that often SAL-based methods give a lower stereotype score, while INLP methods more significantly damage the language model score. This implies that the *SAL-based methods remove bias*

Model	S. Score (%)	LM Score (%)
BERT	60.28	84.17
+ AM + INLP	↓1.14 59.14	↓0.43 83.75
+ KMEANS + INLP	↓0.16 60.12	↓0.47 83.70
+ ORACLEINLP	↓2.93 57.35	↓1.07 83.11
+ PARTIALINLP	↓2.93 57.35	↓1.07 83.10
+ AMSAL	↑0.61 60.89	↑0.09 84.26
+ KMEANS + SAL	↑0.19 60.47	↑0.13 84.30
+ ORACLESAL	↓0.83 59.44	↑0.53 84.70
+ PARTIALSAL	↓0.83 59.44	↑0.53 84.70
ALBERT	59.93	89.77
+ AM + INLP	↓0.29 59.64	↓1.45 88.32
+ KMEANS + INLP	↓0.59 59.34	↓0.08 89.69
+ ORACLEINLP	↓2.73 57.20	↓1.59 88.17
+ PARTIALINLP	↓2.72 57.21	↓1.62 88.15
+ AMSAL	↓0.22 59.71	↓0.32 89.45
+ KMEANS + SAL	↑0.56 60.49	↓0.10 89.67
+ ORACLESAL	↓2.18 57.75	↓0.16 89.61
+ PARTIALSAL	↓2.18 57.75	↓0.16 89.61
RoBERTa	66.32	88.95
+ AM + INLP	↓4.95 61.37	↑0.04 88.99
+ KMEANS + INLP	↓2.20 64.13	↓1.47 87.48
+ ORACLEINLP	↓3.82 62.51	↓0.92 88.03
+ PARTIALINLP	↓3.82 62.51	↓0.91 88.04
+ AMSAL	↓0.63 65.70	↑0.60 89.54
+ KMEANS + SAL	↓0.49 65.83	↑0.46 89.41
+ ORACLESAL	↓3.32 63.00	↑0.40 89.35
+ PARTIALSAL	↓3.32 63.00	↑0.40 89.35
GPT-2	62.65	91.01
+ AM + INLP	↓1.65 61.00	↓3.77 87.24
+ KMEANS + INLP	↓1.57 61.08	↓3.09 87.93
+ ORACLEINLP	↓1.26 61.39	91.01
+ PARTIALINLP	↓1.26 61.39	91.01
+ AMSAL	↓1.58 61.07	↓0.23 90.79
+ KMEANS + SAL	↓4.00 58.64	↓0.60 90.41
+ ORACLESAL	↓4.55 58.09	↓1.75 89.26
+ PARTIALSAL	↓4.55 58.09	↓1.75 89.26

Table 3: StereoSet stereotype scores (Stt. Score) and language modeling scores (LM Score) for the gender category. Stereotype scores indicate the least bias at 50% and the LM scores indicate high usability at 100%.

effectively while less significantly harming the usability of the language model representations.

We also conducted comprehensive results for other datasets (SEAT and GLUE) and categories of bias (based on race and religion). The results, especially for GLUE, demonstrate the effectiveness of our method of unaligned information removal. For GLUE, we consistently retain the

baseline task performance almost in full. See Appendix B.

4.4 Multiple-Guarded Attribute Sentiment

We hypothesize that AM-based methods are better suited for setups where multiple guarded attributes should be removed, as they allow us to target several guarded attributes with different priors. To examine our hypothesis, we experiment with a dataset curated from Twitter (tweets encoded using BERT, `bert-base-uncased`), in which users are surveyed for their age and gender (Cachola et al., 2018). We bucket the age into three groups (0–25, 26–50, and above 50). Tweets in this dataset are annotated with their sentiment, ranging from 1 (very negative) to 5 (very positive). The dataset consists of more than 6,400 tweets written by more than 1,700 users. We removed users who no longer have public Twitter accounts and users with locations that do not exist based on a filter,³ resulting in a dataset with over 3,000 tweets, written by 817 unique users. As tweets are short by nature and their number is relatively small, the debiasing signal in this dataset (the amount of information it contains about the guarded attributes) might not be sufficient for the attribute removal. To amplify this signal, we concatenated each tweet in the dataset to at most ten other tweets from the same user.

We study the relationship between the main task of sentiment detection and the two protected attributes of age and gender. As a protected attribute z , we use the combination of both age and gender as a binary one-hot vector. This dataset presents a use-case for our algorithm of a composed protected attribute. Rather than using a classifier for predicting the sentiment, we use linear regression. Following Cachola et al. (2018), we use Mean Absolute Error (MAE) to report the error of the sentiment predictions. Given that the sentiment is predicted as a continuous value, we cannot use the TPR gap as in previous sections. Rather, we use the following formula:

$$\text{MAEGap} = \text{std}(\text{MAD}_{z=j} \mid j \in [m]), \quad (6)$$

³We used a list of cities, counties, and states in the United States, taken from <https://tinyurl.com/4kmc6pyn>. All users were in the United States when the data was collected by the original curators.

Model	MAE		Age (gap)		Gender (gap)	
BERTMODEL		0.745	0.031		0.011	
+ AM + INLP	↓0.027	0.717	0.031		↓0.008	0.003
+ KMEANS + INLP	↓0.052	0.693	↓0.001	0.030	↑0.010	0.021
+ ORACLEINLP	↓0.022	0.723	↓0.008	0.022	↑0.005	0.017
+ PARTIALINLP	↓0.025	0.719	↑0.007	0.038		0.011
+ AMSAL	↑0.009	0.754	↓0.005	0.026	↓0.009	0.002
+ KMEANS + SAL	↑0.039	0.783	↓0.001	0.030	↓0.007	0.004
+ ORACLESAL	↑0.012	0.757	↓0.002	0.029	↓0.009	0.003
+ PARTIALSAL	↑0.025	0.769	↓0.001	0.030	↓0.005	0.006

Table 4: MAE and debiasing gap values on the Twitter dataset, when using BERT to encode the tweets. For age and gender, we give the MAE gap as in Eq. 5.

where $MAD_{z=j} = \frac{1}{\ell} \sum_i |\eta_{ij} - \mu_j|$ where i ranges over the set of size ℓ of examples with protected attribute value j , μ_j is the average of absolute \mathbf{Y} prediction error for that set and η_{ij} is the absolute difference between μ_j and the absolute error of example i .⁴ The function `std` in this case indicates the standard deviation of the m values of $MAD_{z=j}$, $j \in [m]$.

Results Table 4 presents our results. Overall, AMSAL reduces the gender and age gap in the predictions while not increasing by much MAE. In addition, we can see both AM-based methods outperform their k -means counterparts which increase unfairness (KMEANS + INLP) or significantly harm the downstream-task performance (KMEANS + SAL). We also consider Figure 4, which shows the quality of the assignments of the AM algorithm change as a function of the labeled data used. As expected, the more labeled data we have, the more accurate the assignments are, but the differences are not very large.

4.5 An Example of Our Method Limitations

We now present the main limitation in our approach and setting. This limitation arises when the random variables \mathbf{Y} and \mathbf{Z} are not easily distinguishable through information about \mathbf{X} .

We experiment with a binary sentiment analysis (\mathbf{y}) task, predicted on users’ tweets (\mathbf{x}), aiming to remove information regarding the authors’ ethnic affiliations. To do so, we use a dataset collected by Blodgett et al. (2016), which examined the differences between African-American English speakers and Standard American English

⁴The absolute error of prediction a with true value b is $|a - b|$.

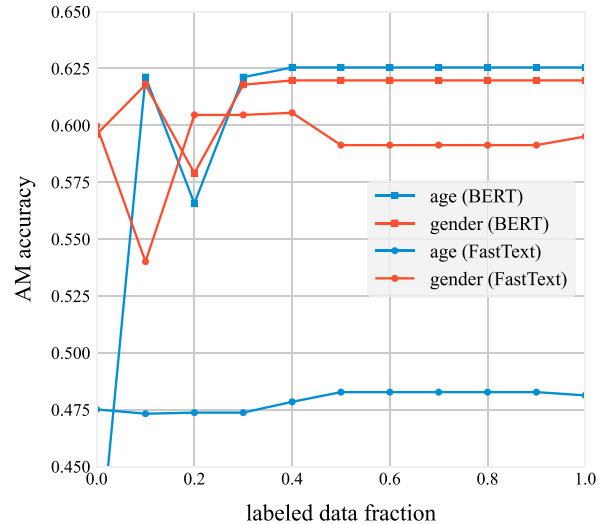


Figure 4: Accuracy of the AM steps with respect to age and gender separately (on unseen data), as a function of the fraction of the labeled dataset used by the AM algorithm.

speakers. As information about one’s ethnicity is hard to obtain, the user’s geolocation information was used to create a distantly supervised mapping between authors and their ethnic affiliations. We follow previous work (Shao et al., 2023; Ravfogel et al., 2020) and use the DeepMoji encoder (Felbo et al., 2017) to obtain representations for the tweets. The train and test sets are balanced regarding sentiment and authors’ ethnicity. We use 20% of the examples for the PARTIAL setting. Table 5 gives the results for this dataset. We observe that the removal with the assignment (k -means, AM, or PARTIAL) significantly harms the performance on the main task and reduces it to a random guess.

This presents a limitation of our algorithm. A priori, there is no distinction between \mathbf{Y} and \mathbf{Z} , as our method is unsupervised. In addition, the positive labels of \mathbf{Y} and \mathbf{Z} have the same prior probability. Indeed, when we check the assignment accuracy in the sentiment dataset, we observe that the k -means, AM, and PARTIAL AM assignment accuracy for identifying \mathbf{Z} are between 0.55 and 0.59. If we check the assignment against \mathbf{Y} , we get an accuracy between 0.74 and 0.76. This means that all assignment algorithms actually identify \mathbf{Y} rather than \mathbf{Z} (both \mathbf{Y} and \mathbf{Z} are binary variables in this case). The conclusion from this is that our algorithm works best when sufficient information on \mathbf{Z} is presented such that it can provide a basis for aligning samples of

Model	Task Acc.	TPR-GAP
DEEPMOJI	0.77	0.14
+ AM + INLP	0.77	0.14
+ KMEANS + INLP	0.77	0.14
+ ORACLEINLP	↓0.02 0.74	↓0.04 0.10
+ PARTIALINLP	↓0.01 0.75	↓0.06 0.08
+ AMSAL	↓0.24 0.52	↑0.03 0.17
+ KMEANS + SAL	↓0.23 0.54	↑0.12 0.26
+ ORACLESAL	0.76	↓0.03 0.11
+ PARTIALSAL	↓0.19 0.57	↑0.15 0.29

Model	F_1 (macro)	TPR-GAP
DEEPMOJI	0.66	0.06
+ AM + INLP	↓0.0002 0.66	0.06
+ KMEANS + INLP	↓0.1 0.56	↓0.02 0.04
+ ORACLEINLP	↓0.19 0.46	↓0.06 0.00
+ PARTIALINLP	↓0.14 0.52	↓0.03 0.03
+ AMSAL	↓0.16 0.49	↓0.02 0.04
+ KMEANS + SAL	↓0.17 0.48	↓0.04 0.02
+ ORACLESAL	↓0.01 0.65	↑0.03 0.09
+ PARTIALSAL	↓0.11 0.54	0.06

Table 5: The performance of removing race information from the DeepMoji dataset is shown for two cases: with balanced ratios of race and sentiment (left) and with ratios of 0.8 for sentiment and 0.5 for race (right). In both cases, the total size of the dataset used is 30,000 examples. To evaluate the performance of the unbalanced sentiment dataset, we use the F_1 macro measure, because in an unbalanced dataset such as this one, a simple classifier that always returns one label will achieve an accuracy of 80%. Such a classifier would have a F_1 macro score of 0.444.

\mathbf{Z} with samples of \mathbf{X} . Suppose such information is unavailable or unidentifiable with information regarding \mathbf{Y} . In that case, we may simply identify the natural clustering of \mathbf{X} according to their main task classes, leading to low main-task performance.

In Table 5, we observe that this behavior is significantly mitigated when the priors over the sentiment and the race are different (0.8 for sentiment and 0.5 for race). In that case, the AM algorithm is able to distinguish between the race-protected attribute (\mathbf{z}) and the sentiment class (\mathbf{y}) quite consistently with INLP and SAL, and the gap is reduced.

We also observe that INLP changed neither the accuracy nor the TPR-GAP for the balanced scenario (Table 5) when using a k -means assignment or an AM assignment. Upon inspection, we found out that INLP returns an identity projection in these cases, unable to amplify the relatively weak signal in the assignment to change the representations.

4.6 Stability Analysis of the Alignment

In Figure 5, we plot the accuracy of the alignment algorithm (knowing the true value of the guarded attribute per input) throughout the execution of the AM steps for the first ten iterations. The shaded area indicates one standard deviation. We observe that the first few iterations are the ones in which the accuracy improves the most. For most of the datasets, the accuracy does not decrease between

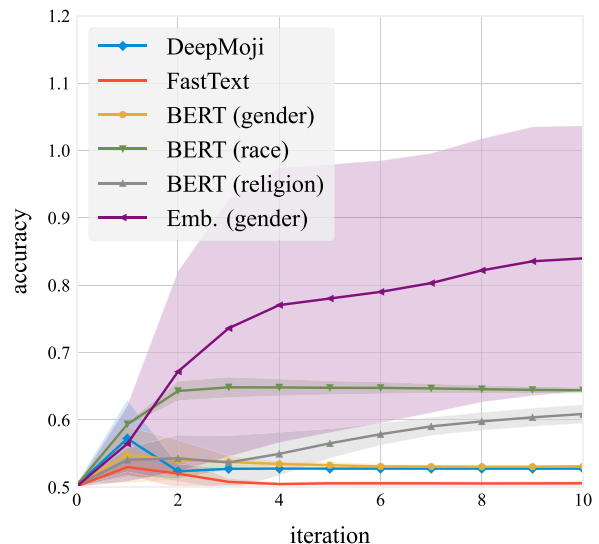


Figure 5: Accuracy of the AM steps (in identifying the correct assignment of inputs to guarded information) as a function of the iteration number. Shaded gray gives upper and lower bound on the standard deviation over five runs with different seeds for the initial π . FastText refers to the BiasBios dataset, the BERT models are for the CrowS-Pairs dataset and Emb. refers to the word embeddings dataset from §4.1.

iterations, though in the case of DeepMoji we do observe a ‘‘bump.’’ This is indeed why the PARTIAL setting of our algorithm, where a small amount of guarded information is available to determine at which iteration to stop the AM algorithm, is important. In the word embeddings case, the variance is larger because, in certain executions, the

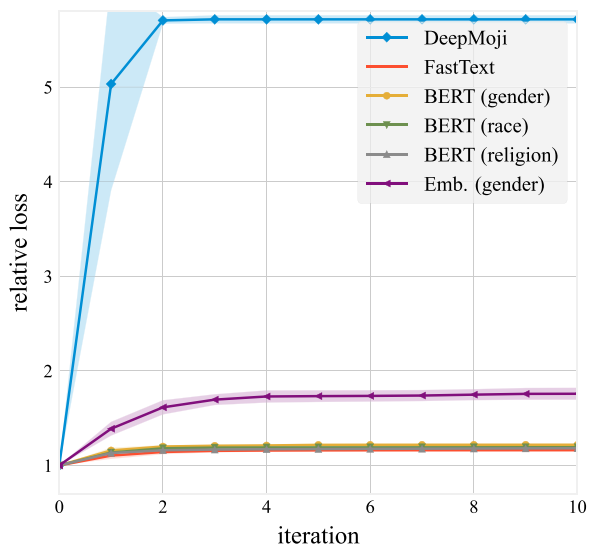


Figure 6: Ratio of the objective value in iteration t and iteration 0 of the ILP for the AM steps as a function of the iteration number t . Shaded gray gives upper and lower bound on the standard deviation over five runs with different seeds for the initial π . See legend explanation in Table 5.

algorithm converged quickly, while in others, it took more iterations to converge to high accuracy.

Figure 6 plots the relative change of the objective value of the ILP from §3.1 against iteration number. The relative change is defined as the ratio between the objective value before the algorithm begins and the same value at a given iteration. We see that there is a relative stability of the algorithm and that the AM steps converge quite quickly. We also observe the DeepMojj dataset has a large increase in the objective value in the first iteration (around $\times 5$ compared to the value the algorithm starts with), after which it remains stable.

5 Related Work

There has been an increasing amount of work on detecting and erasing undesired or protected information from neural representations, with standard software packages for this process having been developed (Han et al., 2022). For example, in their seminal work, Bolukbasi et al. (2016) showed that word embeddings exhibit gender stereotypes. To mitigate this issue, they projected the word embeddings to a neutral space with respect to a ‘‘he-she’’ direction. Influenced by this work, Zhao et al. (2018) proposed a customized training scheme to reduce the gender bias in word em-

beddings. Gonen and Goldberg (2019) examined the effectiveness of the methods mentioned above and concluded they remove bias in a shallow way. For example, they demonstrated that classifiers can accurately predict the gender associated with a word when fed with the embeddings of both debiasing methods.

Another related strand of work uses adversarial learning (Ganin et al., 2016), where an additional objective function is added for balancing undesired-information removal and the main task (Edwards and Storkey, 2016; Li et al., 2018; Coavoux et al., 2018; Wang et al., 2021). Elazar and Goldberg (2018) have also demonstrated that an ad-hoc classifier can easily recover the removed information from adversarially trained representations. Since then, methods for information erasure such as INLP and its generalization (Ravfogel et al., 2020, 2022), SAL (Shao et al., 2023) and methods based on similarity measures between neural representations (Colombo et al., 2022) have been developed. With a similar motivation to ours, Han et al. (2021b) aimed to ease the burden of obtaining guarded attributes at a large scale by decoupling the adversarial information removal process from the main task training. They, however, did not experiment with debiasing representations where no guarded attribute alignments are available. Shao et al. (2023) experimented with the removal of features in a scenario in which a low number of protected attributes is available.

Additional previous work showed that methods based on causal inference (Feder et al., 2021), train-set balancing (Han et al., 2021a), and contrastive learning (Shen et al., 2021; Chi et al., 2022) effectively reduce bias and increase fairness. In addition, there is a large body of work for detecting bias, its evaluation (Dev et al., 2021) and its implications in specific NLP applications. Savoldi et al. (2022) detected a gender bias in speech translation systems for gendered languages. Gender bias is also discussed in the context of knowledge base embeddings by Fisher et al. (2019); Du et al. (2022), and multilingual text classification (Huang, 2022).

6 Conclusions and Future Work

We presented a new and challenging setup for removing information, with minimal or no available sensitive information alignment. This setup

is crucial for the wide applicability of debiasing methods, as for most applications, obtaining such sensitive labels on a large scale is challenging. To ease this problem, we present a method to erase information from neural representations, where the guarded attribute information does not accompany each input instance. Our main algorithm, AMSAL, alternates between two steps (Assignment and Maximization) to identify an assignment between the input instances and the guarded information records. It then completes its execution by removing the information by minimizing covariance between the input instances and the aligned guarded attributes. Our approach is modular, and other erasure algorithms, such as INLP, can be used with it. Experiments show that we can reduce the unwanted bias in many cases while keeping the representations highly useful. Future work might include extending our technique to the kernelized case, analogously to the method of Shao et al. (2023).

Ethical Considerations

The AM algorithm could potentially be misused by, rather than using the AM steps to erase information, using them to link records of two different types, undermining the privacy of the record holders. Such a situation may merit additional concern because the links returned between the guarded attributes and the input instances will likely contain mistakes. The links are unreliable for decision-making at the *individual level*. Instead, they should be used on an aggregate as a statistical construct to erase information from the input representations. Finally,⁵ we note that the automation of the debiasing process, without properly statistically confirming its accuracy using a correct sample, may promote a false sense of security that a given system is making fair decisions. We do not recommend using our method for debiasing without proper statistical control and empirical verification of correctness.

Acknowledgments

We thank the reviewers, the action editors and Marcio Fonseca for their thorough feedback. We also thank Daniel Preoțiuc-Pietro for his help with the Twitter data. We thank Kousha Etessami for

⁵We thank the anonymous reviewer for raising this issue.

being a sounding board for certain parts of the paper. The experiments in this paper were supported by compute grants from the Edinburgh Parallel Computing Center and from the Baskerville Tier 2 HPC service (University of Birmingham).

A Justification of the AM Algorithm: Further Details

We provide here the full details for the claim in §3.5. Our first observation is that for a uniformly sampled permutation $\pi: [n] \rightarrow [n]$, the probability that it has exactly $k \leq n$ elements such that $\pi(i) = i$ for all i in this set of elements is bounded from above by:⁶

$$\frac{\binom{n}{k}(n-k)!}{n!} = \frac{1}{k!}.$$

We also assume that $\mathbb{E}[\mathbf{X} \mid \mathbf{H}] = 0$ and $\mathbb{E}[\mathbf{Z} \mid \mathbf{H}] = 0$, and that the product of every pair of coordinates of \mathbf{X} and \mathbf{Z} is bounded in absolute value by a constant $B > 0$. Let $\{(\mathbf{x}^{(i)}, \mathbf{z}^{(i)}, \mathbf{h}^{(i)})\}$ be a random sample of size n from the joint distribution $p(\mathbf{X}, \mathbf{Z}, \mathbf{H})$. Given a permutation $\pi: [n] \rightarrow [n]$, define $I(\pi) = \{i \mid \pi(i) = i\}$. For a given set $M \subseteq [n]$, define

$$\mathbf{\Omega}_{\pi|M} = \sum_{i \in M} \mathbf{x}^{(i)}(\mathbf{z}^{(\pi(i))})^\top.$$

For a matrix $\mathbf{A} \in \mathbb{R}^{d \times d'}$, let $\sigma_j(\mathbf{A})$ be its j th largest singular value, and let $\sigma^+(\mathbf{A}) = \sum_j \sigma_j(\mathbf{A})$. Let $\sigma^+ = \sigma^+(\mathbb{E}[\mathbf{\Omega}_t])$.

We first note that for any permutation π , it holds that $\mathbb{E}[\mathbf{\Omega}_{\pi|K}] = 0$ where we define $K = [n] \setminus I(\pi)$.

Lemma 1. *For any $t > 0$, it holds that:*

$$p(\|\mathbf{\Omega}_{\pi|I(\pi)} - \mathbb{E}[\mathbf{\Omega}_{\pi|I(\pi)}]\|_2 \geq dd't) \quad (7)$$

is smaller than $2dd' \exp\left(-\frac{t^2}{|I(\pi)|B^2}\right)$.

Proof. By Hoeffding’s inequality, for any $i \in [d]$, $j \in [d']$, it holds that the probability that for

⁶Choose k elements that are fixed, and let the rest vary arbitrarily.

$|I(\pi)|$ i.i.d. random variables $\mathbf{X}^k, \mathbf{Z}^k$ the following is true:

$$\left| \sum_{k \in I(\pi)} \mathbf{X}_i^k \mathbf{Z}_j^k - \sum_{k \in I(\pi)} \mathbb{E}[\mathbf{X}_i^k \mathbf{Z}_j^k] \right| \geq t$$

is smaller than $2 \exp\left(-\frac{t^2}{|I(\pi)|B^2}\right)$. Therefore, by a union bound on each element of the matrix Ω_π , we get the upper bound on Eq. 7. \square

Lemma 2. *For any $t > 0$, it holds that:*

$$\|\Omega_{\pi|K} - \mathbb{E}[\Omega_{\pi|K}]\|_2$$

is smaller than $2|K|dd'B$.

Proof. Since \mathbf{X}_i and \mathbf{Z}_j are bounded as a product in absolute value by B , and the dimensions of $\Omega_{\pi|K}$ is $d \times d'$, each cell being a sum of $|K|$ values, the bound naturally follows. \square

Let n such that $n\sigma^+ > 2kdd'B$ where $k = |K|$. Then from Lemma 2, $\|\Omega_{\pi|K} - \mathbb{E}[\Omega_{\pi|K}]\|_2 < n\sigma^+$. Consider the event $\sigma^+(\Omega_\ell) < \sigma^+(\Omega_\pi)$. Its probability is bounded from above by the probability of the event $\sigma^+(\Omega_\ell) \leq n\sigma^+$ OR $\sigma^+(\Omega_\pi) \geq n\sigma^+$ (for any n as the above). Due to the inequality of Weyl (Theorem 1 in Stewart 1990; see below), the fact that $\Omega_\pi = \Omega_{\pi|K} + \Omega_{\pi|I(\pi)}$, Lemma 1, and the fact that $n - k \leq n$, the probability of this OR event is bounded from above by $4dd' \exp\left(-\frac{(n-k)(\sigma^+)^2}{(dd'B)^2}\right)$.

The conclusion from this is that if we were to sample uniformly a permutation π from the set of permutations over $[n]$, then with quite high likelihood (because the fraction of elements that are preserved under π becomes smaller as n becomes larger), the sum of the singular values of Ω_π under this permutation will be smaller than the sum of the singular values of Ω_ℓ —meaning, when the x s and the z s are correctly aligned. This justifies our objective of aligning the x s and the z s with an objective that maximizes the singular values, following Proposition 1.

Inequality of Weyl (1912) As mentioned by Stewart (1990), the following holds:

Lemma 3. *Let \mathbf{A} and \mathbf{E} be two matrices, and let $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{E}$. Let σ_i be the i th singular value of*

\mathbf{A} and $\tilde{\sigma}_i$ be the i th singular value of $\tilde{\mathbf{A}}$. Then $|\sigma_i - \tilde{\sigma}_i| \leq \|\mathbf{E}\|_2$.

B Comprehensive Results on the BiasBench Datasets

We include more results for the SEAT dataset from BiasBench and for the CrowS-Pairs dataset and StereoSet datasets for bias categories other than gender. A description of the SEAT and GLUE datasets (with metrics used) follows.

SEAT (May et al., 2019) SEAT is a sentence-level extension of WEAT (Caliskan et al., 2017), which is an association test between two categories of words: attribute word sets and target word sets. For example, attribute words for gender bias could be $\{he, man\}$, while a target words could be $\{career, office\}$. For example, an attribute word set (in case of gender bias) could be a set of words such as $\{he, him, man\}$, while a target word set might be words related to office work. If we see a high association between an attribute word set and a target word set, we may claim that a particular gender bias is encoded. The final evaluation is calculated by measuring the similarity between the different attributes and target word sets. To extend WEAT to a sentence-level test, (Caliskan et al., 2017) incorporated the WEAT attribute and target words into synthetic sentence templates.

We use an *effect size* metric to report our results for SEAT. This measure is a normalized difference between cosine similarity of representations of the attribute words and the target words. Both attribute words and target words are split into two categories (for example, in relation to gender), so the difference is based on four terms, between each pair of each category set of words (target and attribute). An effect size closer to zero indicates less bias is encoded in the representations.

GLUE (Wang et al., 2019) We follow Meade et al. (2022) and use the GLUE dataset to test the debiased model on an array of downstream tasks to validate their usability. GLUE is a highly popular benchmark for testing NLP models, containing a variety of tasks, such as classification tasks (e.g., sentiment analysis), similarity tasks (e.g.,

Model	S. Score (%)	LM Score (%)	Model	S. Score (%)	LM Score (%)
Race			Religion		
BERT	57.03	84.17	BERT	59.70	84.17
+ AM + INLP	↑1.23 58.26	↓0.65 83.53	+ AM + INLP	↑3.22 62.92	↓0.60 83.58
+ KMEANS + INLP	↑0.32 57.35	↓0.63 83.54	+ KMEANS + INLP	↑1.67 61.38	↓0.28 83.89
+ ORACLEINLP	↑0.33 57.36	↓1.05 83.12	+ ORACLEINLP	↑0.61 60.31	↓0.82 83.35
+ PARTIALINLP	↑0.33 57.36	↓1.05 83.12	+ PARTIALINLP	↑0.61 60.31	↓0.82 83.35
+ AMSAL	↑1.98 59.01	↑0.55 84.72	+ AMSAL	↑1.42 61.12	↑0.60 84.77
+ KMEANS + SAL	↑2.10 59.13	↑0.49 84.66	+ KMEANS + SAL	↑1.83 61.53	↑0.62 84.79
+ ORACLESAL	↑1.85 58.88	↑0.76 84.93	+ ORACLESAL	↑0.09 59.79	↑0.68 84.85
+ PARTIALSAL	↑1.85 58.88	↑0.76 84.93	+ PARTIALSAL	↑0.09 59.79	↑0.68 84.85
ALBERT	57.57	89.77	ALBERT	60.32	89.77
+ AM + INLP	↓0.94 56.63	↓2.25 87.52	+ AM + INLP	↑1.85 62.17	↓1.20 88.57
+ KMEANS + INLP	↑1.42 59.00	↓2.04 87.72	+ KMEANS + INLP	↑2.78 63.10	↓1.41 88.36
+ ORACLEINLP	↓2.54 55.04	↓1.95 87.82	+ ORACLEINLP	↑3.45 63.77	↓0.91 88.86
+ PARTIALINLP	↓2.54 55.04	↓1.97 87.80	+ PARTIALINLP	↑3.45 63.77	↓0.91 88.86
+ AMSAL	↓0.84 56.73	↑0.09 89.86	+ AMSAL	↓0.78 59.54	↑0.39 90.15
+ KMEANS + SAL	↓1.31 56.27	↑0.06 89.82	+ KMEANS + SAL	↓0.39 59.94	↑0.36 90.13
+ ORACLESAL	↓0.31 57.26	↑0.75 90.52	+ ORACLESAL	↓1.18 59.14	↑1.26 91.02
+ PARTIALSAL	↓0.31 57.26	↑0.75 90.52	+ PARTIALSAL	↓1.18 59.14	↑1.26 91.02
GPT-2	58.83	91.01	GPT-2	63.26	91.01
+ AM + INLP	↓0.90 57.93	↓5.55 85.47	+ AM + INLP	↓1.90 61.36	↓6.57 84.44
+ KMEANS + INLP	↑0.10 58.93	↓0.94 90.07	+ KMEANS + INLP	↑1.66 64.92	↓0.88 90.14
+ ORACLEINLP	↑0.21 59.04	↑0.04 91.06	+ ORACLEINLP	↑0.69 63.95	↑0.19 91.21
+ PARTIALINLP	↑0.21 59.04	↑0.03 91.05	+ PARTIALINLP	↑0.69 63.95	↑0.19 91.21
+ AMSAL	↓3.15 55.69	↓0.43 90.59	+ AMSAL	↓4.83 58.43	↓0.60 90.41
+ KMEANS + SAL	↓2.65 56.18	↑0.08 91.09	+ KMEANS + SAL	↓2.57 60.69	↓0.41 90.60
+ ORACLESAL	↓3.09 55.75	↓2.09 88.92	+ ORACLESAL	↓5.48 57.78	↓3.91 87.10
+ PARTIALSAL	↓3.09 55.75	↓2.09 88.92	+ PARTIALSAL	↓5.48 57.78	↓3.91 87.10
RoBERTa	61.67	88.95	RoBERTa	64.28	88.95
+ AM + INLP	↓7.31 54.37	↓3.06 85.88	+ AM + INLP	↓3.84 60.44	↓4.09 84.86
+ KMEANS + INLP	↓5.67 56.00	↓3.95 85.00	+ KMEANS + INLP	↓1.37 62.91	↓2.82 86.13
+ ORACLEINLP	↓3.42 58.26	↑0.01 88.96	+ ORACLEINLP	↓3.94 60.34	↓0.83 88.12
+ PARTIALINLP	↓3.42 58.26	↑0.02 88.96	+ PARTIALINLP	↓3.94 60.34	↓0.84 88.11
+ AMSAL	↑0.74 62.41	↑1.02 89.96	+ AMSAL	↓1.64 62.64	↑1.00 89.95
+ KMEANS + SAL	↑0.71 62.39	↑1.03 89.97	+ KMEANS + SAL	↓2.04 62.24	↑0.99 89.93
+ ORACLESAL	↑1.79 63.47	↑0.49 89.44	+ ORACLESAL	↓1.92 62.36	↑1.11 90.06
+ PARTIALSAL	↑1.79 63.47	↑0.49 89.44	+ PARTIALSAL	↓1.92 62.36	↑1.11 90.06

Table 6: (a) StereoSet stereotype scores and language modeling scores (LM Score) for race debiased BERT, ALBERT, RoBERTa, and GPT-2 models. Stereotype scores are least biased at 50% and the LM Scores are best at 100%; (b) StereoSet stereotype scores and language modeling scores (LM Score) for religion debiased BERT, ALBERT, RoBERTa, and GPT-2 models. Stereotype scores are least biased at 50% and the LM Scores are best at 100%.

paraphrase identification), and inference tasks (e.g., question-answering).

The following tables of results are included:

- Table 6 presents the StereoSet results for removing the race (a) and religion (b) guarded attributes.

- Tables 7, 8, and 9 describe the SEAT effect sizes for the gender, race, and religion cases, respectively.

- Table 10 presents the scores the debiased representations achieve for the GLUE benchmark.

Model	SEAT6	SEAT6b	SEAT7	SEAT7b	SEAT8	SEAT8b	Avg. Effect Size
BERT	0.931 *	0.090	-0.124	0.937 *	0.783 *	0.858 *	0.620
+ AM + INLP	0.744 *	-0.006	0.036	0.968 *	0.828 *	0.849 *	↓0.049 0.572
+ KMEANS + INLP	0.809 *	0.013	-0.084	0.812 *	0.756 *	0.785 *	↓0.077 0.543
+ ORACLEINLP	0.269	-0.339	-0.403	0.437 *	0.399	0.289	↓0.264 0.356
+ PARTIALINLP	0.269	-0.338	-0.404	0.436 *	0.399	0.289	↓0.265 0.356
+ AMSAL	0.928 *	0.110	-0.191	0.717 *	0.756 *	0.756 *	↓0.044 0.576
+ KMEANS + SAL	0.925 *	0.109	-0.190	0.722 *	0.752 *	0.752 *	↓0.046 0.575
+ ORACLESAL	0.387	-0.301	-0.876	-0.192	0.299	0.309	↓0.227 0.394
+ PARTIALSAL	0.387	-0.301	-0.876	-0.192	0.299	0.309	↓0.227 0.394
ALBERT	0.637 *	0.151	0.487 *	0.956 *	0.683 *	0.823 *	0.623
+ AM + INLP	0.620 *	0.165	0.408 *	0.854 *	0.649 *	0.744 *	↓0.049 0.573
+ KMEANS + INLP	0.645 *	0.147	0.408 *	0.822 *	0.660 *	0.829 *	↓0.038 0.585
+ ORACLEINLP	0.464 *	-0.084	-0.222	0.467 *	0.215	0.462 *	↓0.304 0.319
+ PARTIALINLP	0.464 *	-0.084	-0.222	0.467 *	0.215	0.462 *	↓0.304 0.319
+ AMSAL	0.640 *	0.138	0.477 *	0.933 *	0.666 *	0.820 *	↓0.010 0.612
+ KMEANS + SAL	0.640 *	0.136	0.474 *	0.933 *	0.664 *	0.818 *	↓0.012 0.611
+ ORACLESAL	0.468 *	-0.067	-0.230	0.312	0.305	0.545 *	↓0.302 0.321
+ PARTIALSAL	0.468 *	-0.067	-0.230	0.312	0.305	0.545 *	↓0.302 0.321
RoBERTa	0.922 *	0.208	0.979 *	1.460 *	0.810 *	1.261 *	0.940
+ AM + INLP	0.982 *	0.262	0.845 *	1.575 *	0.840 *	1.395 *	↑0.043 0.983
+ KMEANS + INLP	0.933 *	0.238	1.090 *	1.595 *	1.148 *	1.435 *	↑0.133 1.073
+ ORACLEINLP	0.781 *	0.014	0.651 *	1.281 *	0.708 *	1.160 *	↓0.174 0.766
+ PARTIALINLP	0.782 *	0.014	0.651 *	1.282 *	0.708 *	1.161 *	↓0.174 0.766
+ AMSAL	0.902 *	0.187	1.021 *	1.549 *	0.893 *	1.386 *	↑0.050 0.990
+ KMEANS + SAL	0.920 *	0.182	1.017 *	1.549 *	0.885 *	1.389 *	↑0.051 0.990
+ ORACLESAL	0.695 *	-0.014	0.550 *	1.315 *	0.684 *	1.170 *	↓0.202 0.738
+ PARTIALSAL	0.695 *	-0.014	0.550 *	1.315 *	0.684 *	1.170 *	↓0.202 0.738
GPT-2	0.138	0.003	-0.023	0.002	-0.224	-0.287	0.113
+ AM + INLP	0.141	0.009	-0.017	0.010	-0.213	-0.283	↓0.001 0.112
+ KMEANS + INLP	0.138	0.006	-0.024	0.002	-0.223	-0.287	0.113
+ ORACLEINLP	0.130	-0.005	-0.024	0.000	-0.229	-0.291	0.113
+ PARTIALINLP	0.130	-0.005	-0.024	0.000	-0.229	-0.291	0.113
+ AMSAL	0.280	0.199	0.900 *	0.352	0.408	0.118	↑0.263 0.376
+ KMEANS + SAL	0.298	-0.304	0.030	0.044	0.114	-0.096	↑0.035 0.148
+ ORACLESAL	0.155	-0.379	-0.093	-0.059	-0.039	-0.173	↑0.037 0.150
+ PARTIALSAL	0.155	-0.379	-0.093	-0.059	-0.039	-0.173	↑0.037 0.150

Table 7: SEAT effect sizes for gender-debiased representations of BERT, ALBERT, RoBERTa, and GPT-2 models. Effect sizes closer to 0 are indicative of less biased model representations. Statistically significant effect sizes at $p < 0.01$ are denoted by *. The final column reports the average absolute effect size across all six gender SEAT tests for each debiased model.

Model	ABW-1	ABW-2	SEAT-3	SEAT-3b	SEAT-4	SEAT-5	SEAT-5b	Avg. Effect Size
BERT	-0.079	0.690 *	0.778 *	0.469 *	0.901 *	0.887 *	0.539 *	0.620
+ AM + INLP	0.155	0.583 *	0.769 *	0.341 *	0.889 *	0.937 *	0.403 *	↓0.038 0.582
+ KMEANS + INLP	0.097	0.590 *	0.775 *	0.381 *	0.882 *	0.888 *	0.357 *	↓0.053 0.567
+ ORACLEINLP	0.295	0.565 *	0.799 *	0.369 *	0.977 *	1.039 *	0.432 *	↑0.019 0.639
+ PARTIALINLP	0.295	0.565 *	0.799 *	0.370 *	0.976 *	1.039 *	0.432 *	↑0.019 0.639
+ AMSAL	0.138	0.621 *	0.797 *	0.374 *	0.911 *	1.015 *	0.435 *	↓0.007 0.613
+ KMEANS + SAL	0.131	0.624 *	0.798 *	0.373 *	0.912 *	1.013 *	0.436 *	↓0.008 0.612
+ ORACLESAL	-0.021	0.643 *	0.788 *	0.357 *	0.885 *	0.894 *	0.431 *	↓0.046 0.574
+ PARTIALSAL	-0.021	0.643 *	0.788 *	0.357 *	0.885 *	0.894 *	0.431 *	↓0.046 0.574
ALBERT	-0.014	0.410	1.132 *	-0.252	0.956 *	1.041 *	0.058	0.552
+ AM + INLP	-0.150	0.505 *	1.149 *	-0.244	0.982 *	1.075 *	-0.036	↑0.040 0.592
+ KMEANS + INLP	-0.015	0.484 *	1.162 *	-0.228	0.988 *	1.067 *	-0.033	↑0.017 0.568
+ ORACLEINLP	0.040	0.534 *	1.165 *	-0.150	0.996 *	1.116 *	0.021	↑0.023 0.574
+ PARTIALINLP	0.040	0.534 *	1.165 *	-0.150	0.996 *	1.116 *	0.021	↑0.023 0.574
+ AMSAL	0.283	0.471 *	0.985 *	-0.299	0.802 *	0.938 *	-0.063	↓0.003 0.549
+ KMEANS + SAL	0.292	0.472 *	0.980 *	-0.294	0.799 *	0.935 *	-0.060	↓0.004 0.547
+ ORACLESAL	0.300 *	0.471 *	0.994 *	-0.281	0.813 *	0.949 *	-0.089	↑0.005 0.557
+ PARTIALSAL	0.300 *	0.471 *	0.994 *	-0.281	0.813 *	0.949 *	-0.089	↑0.005 0.557
RoBERTa	0.395 *	0.159	-0.114	-0.003	-0.315	0.780 *	0.386 *	0.307
+ AM + INLP	0.257	0.534 *	0.381 *	0.138	0.202	0.646 *	0.300 *	↑0.044 0.351
+ KMEANS + INLP	0.270	0.466 *	0.242 *	0.116	0.079	0.627 *	0.310 *	↓0.006 0.301
+ ORACLEINLP	0.222	0.445	0.354 *	0.130	0.125	0.636 *	0.301 *	↑0.009 0.316
+ PARTIALINLP	0.222	0.445	0.354 *	0.130	0.125	0.636 *	0.301 *	↑0.009 0.316
+ AMSAL	0.317 *	0.520 *	0.471 *	0.211	0.314 *	0.576 *	0.275 *	↑0.076 0.384
+ KMEANS + SAL	0.314 *	0.522 *	0.476 *	0.212	0.320 *	0.579 *	0.276 *	↑0.078 0.385
+ ORACLESAL	0.170	0.220	0.401 *	0.227	0.225	0.471 *	0.268 *	↓0.024 0.283
+ PARTIALSAL	0.170	0.220	0.401 *	0.227	0.225	0.471 *	0.268 *	↓0.024 0.283
GPT-2	1.060 *	-0.200	0.431 *	0.243 *	0.133	0.696 *	0.370 *	0.448
+ AM + INLP	1.046 *	-0.169	0.472 *	0.257 *	0.172	0.686 *	0.366 *	↑0.005 0.452
+ KMEANS + INLP	1.059 *	-0.189	0.440 *	0.248 *	0.141	0.695 *	0.373 *	↑0.002 0.449
+ ORACLEINLP	1.060 *	-0.200	0.433 *	0.246 *	0.135	0.693 *	0.364 *	0.447
+ PARTIALINLP	1.060 *	-0.200	0.433 *	0.246 *	0.135	0.693 *	0.364 *	0.447
+ AMSAL	-0.525	0.157	0.654 *	0.398 *	0.327 *	0.120	0.117	↓0.119 0.328
+ KMEANS + SAL	-0.459	0.179	1.088 *	0.499 *	0.827 *	0.594 *	0.271 *	↑0.112 0.560
+ ORACLESAL	0.183	0.101	1.095 *	0.515 *	0.836 *	0.817 *	0.347 *	↑0.109 0.556
+ PARTIALSAL	0.183	0.101	1.095 *	0.515 *	0.836 *	0.817 *	0.347 *	↑0.109 0.556

Table 8: SEAT effect sizes for race debiased BERT, ALBERT, RoBERTa, and GPT-2 models. Effect sizes closer to 0 are indicative of less biased model representations. Statistically significant effect sizes at $p < 0.01$ are denoted by *. The final column reports the average absolute effect size across all six gender SEAT tests for each debiased model.

Model	Religion-1	Religion-1b	Religion-2	Religion-2b	Avg. Effect Size
BERT	0.744 *	-0.067	1.009 *	-0.147	0.492
+ AM + INLP	0.530 *	-0.184	0.847 *	-0.156	↓0.062 0.429
+ KMEANS + INLP	0.574 *	-0.253	0.919 *	-0.308	↑0.022 0.514
+ ORACLEINLP	0.473 *	-0.301	0.787 *	-0.280	↓0.031 0.460
+ PARTIALINLP	0.473 *	-0.301	0.787 *	-0.280	↓0.031 0.460
+ AMSAL	0.683 *	-0.117	0.941 *	-0.178	↓0.012 0.480
+ KMEANS + SAL	0.686 *	-0.112	0.938 *	-0.180	↓0.013 0.479
+ ORACLESAL	0.735 *	-0.036	0.884 *	-0.156	↓0.039 0.453
+ PARTIALSAL	0.735 *	-0.036	0.884 *	-0.156	↓0.039 0.453
ALBERT	0.203	-0.117	0.848 *	0.555 *	0.431
+ AM + INLP	0.208	-0.065	0.891 *	0.557 *	↓0.001 0.430
+ KMEANS + INLP	0.126	-0.138	0.839 *	0.518 *	↓0.025 0.405
+ ORACLEINLP	0.206	-0.110	0.727 *	0.385 *	↓0.074 0.357
+ PARTIALINLP	0.206	-0.110	0.727 *	0.385 *	↓0.074 0.357
+ AMSAL	0.024	-0.256	0.722 *	0.418 *	↓0.076 0.355
+ KMEANS + SAL	0.027	-0.253	0.722 *	0.415 *	↓0.077 0.354
+ ORACLESAL	0.116	-0.168	0.585 *	0.289	↓0.141 0.289
+ PARTIALSAL	0.116	-0.168	0.585 *	0.289	↓0.141 0.289
RoBERTa	0.132	0.018	-0.191	-0.166	0.127
+ AM + INLP	-0.042	-0.203	-0.255	-0.273	↑0.067 0.193
+ KMEANS + INLP	-0.014	-0.204	-0.187	-0.304	↑0.051 0.177
+ ORACLEINLP	-0.309	-0.347	-0.191	-0.135	↑0.119 0.246
+ PARTIALINLP	-0.309	-0.347	-0.191	-0.135	↑0.119 0.246
+ AMSAL	-0.169	-0.228	-0.014	0.009	↓0.022 0.105
+ KMEANS + SAL	-0.172	-0.231	-0.011	0.011	↓0.020 0.106
+ ORACLESAL	-0.063	-0.208	-0.203	-0.109	↑0.019 0.146
+ PARTIALSAL	-0.063	-0.208	-0.203	-0.109	↑0.019 0.146
GPT-2	-0.332	-0.271	0.617 *	0.286	0.376
+ AM + INLP	-0.326	-0.264	0.671 *	0.333	↑0.022 0.399
+ KMEANS + INLP	-0.331	-0.271	0.624 *	0.297	↑0.004 0.380
+ ORACLEINLP	-0.331	-0.271	0.615 *	0.284	↓0.001 0.375
+ PARTIALINLP	-0.331	-0.271	0.615 *	0.284	↓0.001 0.375
+ AMSAL	0.087	0.064	0.767 *	0.341	↓0.062 0.315
+ KMEANS + SAL	0.274	0.300	1.014 *	0.534 *	↑0.154 0.531
+ ORACLESAL	-0.101	-0.089	1.144 *	0.779 *	↑0.152 0.528
+ PARTIALSAL	-0.101	-0.089	1.144 *	0.779 *	↑0.152 0.528

Table 9: SEAT effect sizes for religion debiased BERT, ALBERT, RoBERTa, and GPT-2 models. Effect sizes closer to 0 are indicative of less biased model representations. Statistically significant effect sizes at $p < 0.01$ are denoted by *. The final column reports the average absolute effect size across all six gender SEAT tests for each debiased model.

Model	cola	mnli	mrpc	qnli	qqp	rte	sst2	stsb	wnli	Average
BERT	56.50	84.73	87.67	91.35	91.00	64.38	92.55	88.51	44.60	77.92
+ AM + INLP	57.69	84.67	88.75	91.24	90.88	64.38	92.74	88.80	42.25	↑0.01 77.93
+ KMEANS + INLP	56.84	84.72	88.23	91.35	90.94	64.02	92.62	88.61	40.38	↓0.40 77.52
+ ORACLEINLP	57.44	84.62	88.32	91.37	91.02	63.66	92.74	88.32	46.48	↑0.30 78.22
+ PARTIALINLP	57.36	84.68	88.09	91.36	91.05	65.46	92.47	88.74	43.19	↑0.12 78.04
+ AMSAL	57.19	84.85	88.48	91.28	90.96	63.78	92.66	88.75	37.56	↓0.64 77.28
+ KMEANS + SAL	56.94	84.76	88.68	91.43	90.91	64.02	92.66	88.74	37.56	↓0.62 77.30
+ ORACLESAL	56.68	84.72	88.18	91.24	90.93	64.50	92.74	88.67	41.78	↓0.20 77.72
+ PARTIALSAL	56.16	84.71	87.82	91.31	90.92	64.50	92.58	88.70	41.78	↓0.31 77.61
ALBERT	46.55	85.31	91.17	91.73	90.82	70.88	91.55	90.57	43.66	78.03
+ AM + INLP	57.30	85.46	90.57	91.63	90.49	70.28	91.86	90.62	48.83	↑1.65 79.67
+ KMEANS + INLP	55.91	85.59	90.74	91.70	90.62	68.71	91.86	90.94	46.48	↑1.15 79.17
+ ORACLEINLP	55.43	85.28	91.27	91.65	90.80	72.08	91.86	90.76	40.38	↑0.81 78.83
+ PARTIALINLP	56.27	85.43	91.39	91.58	90.68	71.60	92.51	90.70	46.01	↑1.55 79.57
+ AMSAL	55.51	85.28	91.35	91.36	90.62	73.41	91.67	90.61	38.50	↑0.67 78.70
+ KMEANS + SAL	55.45	85.54	91.33	91.53	90.66	73.16	91.97	90.69	39.91	↑0.89 78.92
+ ORACLESAL	54.50	85.44	92.09	91.78	90.70	72.68	92.05	90.79	43.19	↑1.22 79.25
+ PARTIALSAL	56.68	85.37	91.31	91.51	90.80	69.19	92.35	90.59	43.19	↑0.98 79.00
RoBERTa	58.38	87.63	92.06	92.64	91.28	71.12	94.15	90.22	52.58	81.12
+ AM + INLP	57.91	87.58	91.66	92.57	91.31	71.48	94.15	90.10	52.11	↓0.13 80.99
+ KMEANS + INLP	57.37	87.51	91.47	92.83	91.26	69.92	94.15	89.96	56.34	↑0.08 81.20
+ ORACLEINLP	56.26	87.50	91.85	92.75	91.34	72.08	94.57	90.12	52.11	↓0.16 80.95
+ PARTIALINLP	57.53	87.70	92.26	92.68	91.31	69.80	94.19	90.15	56.34	↑0.21 81.33
+ AMSAL	57.47	87.63	91.50	92.81	91.31	69.19	94.30	90.01	56.34	↑0.06 81.17
+ KMEANS + SAL	58.23	87.56	92.51	92.97	91.31	69.92	94.00	89.96	53.99	↑0.04 81.16
+ ORACLESAL	58.47	87.85	92.22	92.84	91.27	70.76	94.50	89.94	52.11	↓0.01 81.11
+ PARTIALSAL	58.05	87.52	91.25	92.95	91.23	69.19	94.27	90.06	52.11	↓0.38 80.74
GPT-2	32.73	82.69	84.06	87.80	89.20	65.46	92.16	84.50	40.85	73.27
+ AM + INLP	33.78	82.66	83.91	87.77	89.15	64.86	92.13	84.47	40.85	↑0.01 73.29
+ KMEANS + INLP	32.67	82.68	84.36	87.83	89.16	65.46	92.13	84.51	40.85	↑0.02 73.29
+ ORACLEINLP	33.00	82.67	84.16	87.82	89.19	65.46	92.20	84.51	40.38	↓0.01 73.27
+ PARTIALINLP	34.51	82.56	84.22	87.88	89.17	65.34	91.78	84.10	40.85	↑0.11 73.38
+ AMSAL	35.07	82.77	84.86	88.21	89.13	65.94	92.28	83.93	40.38	↑0.35 73.62
+ KMEANS + SAL	35.57	82.78	84.80	88.27	89.16	65.34	92.13	83.93	38.97	↑0.17 73.44
+ ORACLESAL	37.23	82.77	84.64	88.39	89.14	64.50	92.05	84.02	40.85	↑0.46 73.73
+ PARTIALSAL	37.66	82.71	85.20	88.28	89.21	66.55	92.13	84.15	39.91	↑0.71 73.98

Table 10: GLUE tests for gender-debiased BERT, ALBERT, RoBERTa, and GPT-2 models.

References

- Su Lin Blodgett, Lisa Green, and Brendan O'Connor. 2016. Demographic dialectal variation in social media: A case study of African-American English. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1119–1130, Austin, Texas. Association for Computational Linguistics. <https://doi.org/10.18653/v1/D16-1120>
- Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Tauman Kalai. 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5–10, 2016, Barcelona, Spain*, pages 4349–4357.
- Isabel Cachola, Eric Holgate, Daniel Preoțiu-Pietro, and Junyi Jessy Li. 2018. Expressively vulgar: The socio-dynamics of vulgarity and its effects on sentiment analysis in social media. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2927–2938, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186. <https://doi.org/10.1126/science.aal4230>, PubMed: 28408601
- Jianfeng Chi, William Shand, Yaodong Yu, Kai-Wei Chang, Han Zhao, and Yuan Tian. 2022. Conditional supervised contrastive learning for fair text classification. *ArXiv preprint*, abs/2205.11485.
- Maximin Coavoux, Shashi Narayan, and Shay B. Cohen. 2018. Privacy-preserving neural representations of text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1–10, Brussels, Belgium. Association for Computational Linguistics. <https://doi.org/10.18653/v1/D18-1001>
- Pierre Colombo, Guillaume Staerman, Nathan Noiry, and Pablo Piantanida. 2022. Learning disentangled textual representations via statistical measures of similarity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2614–2630, Dublin, Ireland. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.acl-long.187>
- Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnamurthy Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 120–128. <https://doi.org/10.1145/3287560.3287572>
- Sunipa Dev, Tao Li, Jeff M. Phillips, and Vivek Srikumar. 2021. OSCaR: Orthogonal subspace correction and rectification of biases in word embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5034–5050, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2021.emnlp-main.411>
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Yupei Du, Qi Zheng, Yuanbin Wu, Man Lan, Yan Yang, and Meirong Ma. 2022. Understanding gender bias in knowledge base embeddings. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1381–1395, Dublin, Ireland. Association for Computational Linguistics.

- Harrison Edwards and Amos J. Storkey. 2016. Censoring representations with an adversary. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2–4, 2016, Conference Track Proceedings*.
- Yanai Elazar and Yoav Goldberg. 2018. Adversarial removal of demographic attributes from text data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 11–21, Brussels, Belgium. Association for Computational Linguistics. <https://doi.org/10.18653/v1/D18-1002>
- Amir Feder, Nadav Oved, Uri Shalit, and Roi Reichart. 2021. CausaLM: Causal model explanation through counterfactual language models. *Computational Linguistics*, 47(2):333–386. <https://doi.org/10.1162/colia.00404>
- Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan, and Sune Lehmann. 2017. Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1615–1625, Copenhagen, Denmark. Association for Computational Linguistics. <https://doi.org/10.18653/v1/D17-1169>
- Joseph Fisher, Dave Palfrey, Christos Christodoulopoulos, and Arpit Mittal. 2019. Measuring social bias in knowledge graph embeddings. *ArXiv preprint*, abs/1912.02761. <https://doi.org/10.18653/v1/2020.emnlp-main.595>
- Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.
- Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 609–614, Minneapolis, Minnesota. Association for Computational Linguistics.
- Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021a. Balancing out bias: Achieving fairness through training reweighting. *ArXiv preprint*, abs/2109.08253.
- Xudong Han, Timothy Baldwin, and Trevor Cohn. 2021b. Decoupling adversarial training for fair NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 471–477, Online. Association for Computational Linguistics.
- Xudong Han, Aili Shen, Yitong Li, Lea Frermann, Timothy Baldwin, and Trevor Cohn. 2022. fairlib: A unified framework for assessing and improving classification fairness. *ArXiv preprint*, abs/2205.01876.
- Xiaolei Huang. 2022. Easy adaptation to mitigate gender bias in multilingual text classification. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 717–723, Seattle, United States. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.naacl-main.52>
- Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Herve Jégou, and Tomas Mikolov. 2016. Fasttext.zip: Compressing text classification models. *ArXiv preprint*, abs/1612.03651.
- Harold W. Kuhn. 1955. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97. <https://doi.org/10.1002/nav.3800020109>
- Yitong Li, Timothy Baldwin, and Trevor Cohn. 2018. Towards robust and privacy-preserving text representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 25–30, Melbourne, Australia. Association for Computational Linguistics.
- David J. C. MacKay. 2003. *Information theory, Inference and Learning Algorithms*. Cambridge University Press.
- Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence

- encoders. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 622–628, Minneapolis, Minnesota. Association for Computational Linguistics.
- Nicholas Meade, Elinor Poole-Dayyan, and Siva Reddy. 2022. An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1878–1898, Dublin, Ireland. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.acl-long.132>
- Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. StereoSet: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5356–5371, Online. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2021.acl-long.416>
- Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R. Bowman. 2020. CrowS-pairs: A challenge dataset for measuring social biases in masked language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1953–1967, Online. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.emnlp-main.154>
- Lyle Ramshaw and Robert E. Tarjan. 2012. On minimum-cost assignments in unbalanced bipartite graphs. HP Labs, Palo Alto, CA, USA, Technical Report HPL-2012-40R1.
- Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7237–7256, Online. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2020.acl-main.647>
- Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D. Cotterell. 2022. Linear adversarial concept erasure. In *International Conference on Machine Learning*, pages 18400–18421. PMLR.
- Beatrice Savoldi, Marco Gaido, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2022. Under the morphosyntactic lens: A multifaceted evaluation of gender bias in speech translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1807–1824, Dublin, Ireland. Association for Computational Linguistics. <https://doi.org/10.18653/v1/2022.acl-long.127>
- Shun Shao, Yftah Ziser, and Shay B. Cohen. 2023. Gold doesn’t always glitter: Spectral removal of linear and nonlinear guarded attribute information. In *Proceedings of the 17th Annual Meeting of the European chapter of the Association for Computational Linguistics (EACL)*, volume abs/2203.07893.
- Aili Shen, Xudong Han, Trevor Cohn, Timothy Baldwin, and Lea Frermann. 2021. Contrastive learning for fair representations. *ArXiv preprint*, abs/2109.10645.
- Gilbert W. Stewart. 1990. Perturbation theory for the singular value decomposition, Technical Report UMIACS-90-120 / CS-TR 2539, University of Maryland, College Park.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6–9, 2019*. OpenReview.net.
- Liwen Wang, Yuanmeng Yan, Keqing He, Yanan Wu, and Weiran Xu. 2021. Dynamically disentangling social bias from task-oriented representations with adversarial attack. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3740–3750, Online. Association

for Computational Linguistics. <https://doi.org/10.18653/v1/2021.naacl-main.293>
Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning gender-neutral word embeddings. In *Proceed-*

ings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4847–4853, Brussels, Belgium. Association for Computational Linguistics. <https://doi.org/10.18653/v1/D18-1521>