Iterative Adversarial Removal of Gender Bias in Pretrained Word Embeddings

Yacine Gaci, Boualem Benatallah, Fabio Casati, Khalid Benabdeslem

Iterative Adversarial Removal of Gender Bias in Pretrained Word Embeddings

Yacine Gaci[1], Boualem Benatallah[2], Fabio Casati[3,4], Khalid Benabdeslem[1]
1 LIRIS - University of Lyon, France
{yacine.gaci, khalid.benabdeslem}@univ-lyon1.fr
2 University of New South Wales, Australia
b.benatallah@unsw.edu.au
3 University of Trento, Italy
4 ServiceNow, USA
fabio.casati@gmail.com

ABSTRACT

Recent advances in Representation Learning have discovered a strong inclination for pre-trained word embeddings to demonstrate unfair and discriminatory gender stereotypes. These usually come in the shape of unjustified associations between representations of group words (e.g., male or female) and attribute words (e.g. driving, cooking, doctor, nurse, etc.) In this paper, we propose an iterative and adversarial procedure to reduce gender bias in word vectors. We aim to remove gender influence from word representations that should otherwise be free of it, while retaining meaningful gender information in words that are inherently charged with gender polarity (male or female). We confine these gender signals in a sub-vector of word embeddings to make them more interpretable. Quantitative and qualitative experiments confirm that our method successfully reduces gender bias in pre-trained word embeddings with minimal semantic offset.

CCS CONCEPTS

• **Computing methodologies** → **Lexical semantics**; **Learning latent representations**; • **Security and privacy** → *Social aspects of security and privacy*;

KEYWORDS

reducing gender bias, word embeddings, adversarial training

ACM Reference Format:
Yacine Gaci[1], Boualem Benatallah[2], Fabio Casati[3,4], Khalid Benabdeslem[1]. 2022. Iterative Adversarial Removal of Gender Bias in Pretrained Word Embeddings. In *The 37th ACM/SIGAPP Symposium on Applied Computing (SAC '22), April 25–29, 2022, Virtual Event, .* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3477314.3507274

1 INTRODUCTION

Word embedding models have met tremendous success in the Representation Learning community thanks to their ability to automatically learn effective semantic features for the predictive task at hand [26, 29]. However, they allow next to no human control into their learned representations, taking directions exclusively from the training data. This fogs our attempts at trying to figure out what is truly encoded in these embeddings, thus the challenge of understanding how embedding models arrange the semantic pieces of language remains wide open. Fortunately, recent studies started to expose undesirable patterns that word embeddings inherit from textual data [2, 5]. One of these patterns is that embedding models exhibit significant levels of sexist, racist, unfair and discriminatory biases [5]. For instance, Bolukbasi et al. [2] found that occupation words such as *doctor*, *lawyer* and *programmer* are much closer in the vector space to male than female words, whereas occupations such as *nurse* and *receptionist* display the opposite behavior.

In addition to the representational harm [1] brought forward by these biased embeddings, the inconvenient effects of stereotyping seep into the downstream applications in which these word vectors are used. Zhao et al. [39] and Rudinger et al. [32] show that co-reference resolution systems rely on stereotypical associations for their predictions. Stanovsky et al. [35] find that machine translation models are sexist due to the underlying word embeddings. The effects of gender bias are perpetuated when these biased word embedding models are used in high-stakes settings such as resume filtering systems which may discriminate against some candidates based on gender alone, as reflected in their names; or job recommendation systems which may rank male applicants higher than their female competitors.

Given the wide adoption of word embeddings and the serious threat they pose toward fairness across gender, there is an urgent need to debias them before they are applied to downstream NLP tasks. In this spirit, we propose a method to reduce *binary* gender bias[1] in word embeddings based on adversarial learning [11]. We can classify existing debiasing methods as projection-based [2, 24], encoding-based [14] or adversarial learning-based [22, 38] approaches. The first class of methods, pioneered by the

[1]We consider two classes in binary gender: male and female. Although we acknowledge that this definition does not reflect the wider scope of gender, and recognize that there are many important ethical design principles and considerations when studying human beings in NLP [19], in this paper, we follow existing research and limit our study to binary gender

work of Bolukbasi et al. [2] and later expanded by others (e.g. [17, 24, 30, 36]) debias word vectors by making them orthogonal to a pre-constructed gender direction through *linear* projections. The main limitation of these methods is that debiasing is linear. Thus, they might miss hints of gender bias that are manifested in non-linear representations. We propose a new bias reduction scheme capable of recognizing non-linear bias forms. Second, encoding-based approaches [14, 15] employ autoencoders to learn debiased latent representations, that they use to reconstruct the original embeddings for the preservation of semantic properties, as per the autoencoder requirements [33]. This objective introduces a strain to the debiasing pipeline because in one hand the latent representations must be free of any gender bias influence. On the other hand, they must encode enough of it to be able to reconstruct the original embeddings. We also use an autoencoder in this work, but we learn two latent representations instead of one. We map each word vector w to two sub-vectors $w^{(g)}$ and $w^{(a)}$; the former must capture all gender information while the latter must be free of it. We provide the decoder with both sub-vectors in order to reconstruct the original embedding w. Therefore, we can focus on debiasing $w^{(a)}$ without worrying too much about correct reconstruction since gender information is already confined in $w^{(g)}$ and given to the decoder.

Finally, adversarial training approaches have long been used in the literature to remove sensitive information from neural representations [22, 23, 38]. However, research shows that although it is possible to hide sensitive information (gender in our case) from an adversary during training, another adversary trained post-hoc can still recover most, if not all, cues about the protected sensitive attribute [9]. Elazar and Goldberg [9] argue that adversarial training alone is not enough for such a task. To overcome this problem, we propose an iterative method for debiasing word embeddings, where we train a new adversary in each iteration, and encourage the embedding model to learn how to fool these adversaries, such that gender bias information is incrementally distilled from different perspectives, a few bits at a time. By the end of this iterative adversarial procedure, we would learn to map each word vector into two coherent sub-vectors: $w^{(g)}$ which encodes gender, and $w^{(a)}$ which is free of it. Debiasing becomes thus straightforward by setting the $w^{(g)}$ component of gender-free words to 0, and using the decoder to go back to the original embedding space. We use existing lexical dictionaries as external knowledge bases to decide which words to debias.

To summarize, we make the following contributions:

- We propose a new post-processing method for reducing binary gender bias in word embeddings. Our method is iterative and reduces bias incrementally in each iteration[2].
- We make word vectors more interpretable by confining gender information into a subset of the embedding model's dimensions.
- We evaluate our method using a stack of qualitative and quantitative experiments aiming to assess both stereotypical and semantic properties of the resulting embeddings.

[2]We release our code and data at https://github.com/YacineGACI/ADV-Debias

2 RELATED WORK

2.1 Bias Detection in NLP systems

A lot of research has been directed toward studying the nature of social stereotypes in word embeddings. Caliskan et al. [5] found that popular word embedding models such as Glove [29] recover a wide array of stereotypical associations from the data it has been trained on. They introduced the Word Embedding Association Test (WEAT) which is a statistical permutation test for measuring bias in word vectors given sets of group and attribute terms. They revealed a strong inclination of word vectors to encode prejudice and biases in their embedded semantics, and paved the path for extensive research to be conducted in the field of stereotype in Natural Language Processing (NLP). Similar to WEAT test, May et al. [25] introduced Sentence Embedding Association Test (SEAT) which generalizes WEAT to sentence embeddings, also using cosine similarity to detect biases. Kurita et al. [18] took another approach for looking at bias by investigating differences in word likelihoods in language models. For example, given the masked sentence *"[MASK] is a doctor"*, if the language model provides different probabilities for the words *he* and *she* to fill in the mask, this means that the language model exhibits gender bias in the first place. Also on this lead, StereoSet [27] and Crows-Pairs [28] are two recent benchmarks to quantify the extent of different types of prejudice (gender, race, religion, age, sexual orientation...) in language models.

Apart from investigating social biases at representation level as depicted in the embeddings, other works [7, 21, 32, 34, 35, 39] probed downstream NLP tasks for their tendency to encode harmful stereotypes. In co-reference resolution systems, Zhao et al. [39] introduced a new benchmark to test for gender bias, an found that current co-reference resolution systems are prejudiced in associating certain occupation words to one gender at the detriment of the other. Parallel to this work, Rudinger et al. [32] proposed another benchmark for gender bias, and experimented with three different types of co-reference resolution systems: rule-based, statistical and neural, finding them all to behave contrary to a gender-neutral fashion. Similarly, other works identified bias manifestations in machine translation systems [35], language inference [7, 8], question answering [21] and automatic language generation [34].

Independently, Brunet et al. [3] developed a methodology based on influence functions [6, 16] to address the question of understanding how biases arise during training. Their proposed technique perturbs the training data and quantifies the difference of bias in the resulting embedding model. Interestingly, their approach allows to trace the origins of bias back to the original training data. In doing so, one can remove the subsets of data leading to the most dramatic bursts of bias. However, their method is costly and time-consuming as it involves retraining the word embedding models from scratch. In this work, we also aim to reduce gender bias in word embeddings. The difference is that our method does not assume retraining, but alters already existing word vectors such that bias is minimized without too much harm to the general semantics.

2.2 Bias Reduction in Word Embeddings

A major part of NLP research community focused on reducing gender stereotypes. Bolukbasi et al. [2] manually determined the vector

direction that captures most of gender information in the embedding space by taking the first principal component of difference vectors relating to gendered pairs (e.g., $\overrightarrow{he} - \overrightarrow{she}$, $\overrightarrow{man} - \overrightarrow{woman}$, $\overrightarrow{boy} - \overrightarrow{girl}$...). They proposed two post-processing debiasing strategies: *Hard-Debias* which projects gender-neutral words onto a subspace that is orthogonal to the gender direction, and *Soft-Debias* which applies a linear transformation that (1) preserves pairwise inner products between word vectors, and (2) minimizes the projection of gender-neutral words on the gender direction. Both *Hard-Debias* and *Soft-Debias* require identifying which words in the vocabulary are neutral with regards to gender and should therefore be debiased. The authors of [2] train a Support Vector Machine (SVM) for debiasing decisions. Therefore, if the SVM predicts a word not to be gender-neutral, it will not get debiased.

In the same spirit, a myriad of other works [14, 17, 24, 36], utilized the notion of gender direction for debiasing pre-trained word vectors. Manzini et al. [24] generalized the work of [2] to cater for multiclass bias types such as race, religion or non-binary gender by identifying bias subspaces rather than bias directions. Wang et al. [36] argue that discrepancies in word frequency significantly impact the geometry of word embeddings and can twist the gender direction. Consequently, they propose to project word embeddings into an intermediate subspace by subtracting components related to word frequency before they apply the pipeline described in [2]. Similarly, Ravfogel et al. [30] suggest a data-driven approach to learn a set of gender directions on which to project word embeddings. Instead of relying on gendered word lists, they train a linear classifier and iteratively project word vectors on the null space of the classifier's matrix.

Also relying on linear projections but taking another approach, Kumar et al. [17] alter the spatial distributions of word embeddings with attraction and repulsion mechanisms. The intuition behind repulsion is that words which are clustered together due to stereotypical constructs must be disassociated. For example, $\overrightarrow{nurse}$ and $\overrightarrow{receptionist}$ are semantically dissimilar but stereotypically close to each other because they are both believed to be more feminine than masculine. Consequently, they have to be repulsed from each other. Attraction on the other hand, minimizes the loss of semantic information by attracting each word to its new vector. Kaneko and Bollegala [14] train an autoencoder to learn latent word representations that keep gender information for gender-definitional words (male or female) but remove it from gender-neutral words. We also use an autoencoder in our work to do the same. However, we reduce gender bias in a non-linear fashion through training non-linear adversaries to recognize gender, and then adjusting the autoencoder to fool these adversaries.

Unlike the aforementioned post-processing approaches, Zhao et al. [40] proposed Gender-Neutral Global Vectors (GN-GloVe) by adding a new constraint to GloVe's objective function such that gender information is confined in a sub-vector. GN-GloVe maximizes the l_2 distance between gendered sub-vectors while it minimizes GloVe's original objective. This method trains word embeddings from scratch and cannot be used to debias existing embedding models. We use an equivalent trick to steer gender information into a subset of the vector dimensions while we encourage the remaining dimensions to be free of gender influence with multiple adversaries.

3 ITERATIVE ADVERSARIAL DEBIASING OF WORD EMBEDDINGS

3.1 Overview

In this work, we choose binary gender as the bias type to mitigate. Although we use GloVe [29] in this paper, our solution neither assumes knowledge about the learning algorithm of the underlying embedding model nor the linguistic resources with which pretraining has been conducted. Thus, our method can be applied *off-the-shelf* on other static embedding models.

Given a pretrained set of d-dimensional word embeddings $\{w^i\}_{i=1}^{|\mathcal{V}|}$ over a vocabulary $\mathcal{V}$, our goal is to learn a transformation E: $\mathbb{R}^d \rightarrow \mathbb{R}^{a+g}$ that projects the original word embeddings into a latent space where gender information is controlled and word semantics are minimally altered. In this new space, a word vector w comprises two parts: $w^{(a)} \in \mathbb{R}^a$ and $w^{(g)} \in \mathbb{R}^g$ such that $w^{(g)}$ monopolizes gender information whereas $w^{(a)}$ should be devoid of any hint about gender. In this case, g is the number of dimensions reserved for gender information[3].

In this section, we give a high-level description of our approach. We use an autoencoder with multiple adversaries to train our debiasing model. The encoder part E projects an input word vector into two separate representations $w^{(a)}$ and $w^{(g)}$ as discussed before. We remove all gender information from $w^{(a)}$ by first training a non-linear classifier (that we call C_1) to classify the gender of a word given its $w^{(a)}$ component. Intuitively, if the classifier is able to correctly recognize gender, we can assume that gender information is still rife within $w^{(a)}$. For this reason, we finetune the autoencoder in the following step to produce latent representations for $w^{(a)}$ such that C_1 is unable to correctly predict gender. In other words, we train the autoencoder in an adversarial way to fool C_1 and prevent it from accessing gender information.

Most previous works based on adversarial training stop at fooling one classifier [9, 23]. However, as discussed in the Introduction, even though C_1 has chance-level accuracy in predicting gender, it is still possible to train another classifier C_2 capable of drawing pretty good decision boundaries when it comes to predicting gender in the new space. This limitation owes to the fact that the adversarial setup discussed so far compels the autoencoder to change its encodings to fool C_1 exclusively, but not every gender classifier. It is likely that gender information is still hiding in $w^{(a)}$, only inaccessible to C_1, but potentially easily recoverable by other classifiers. Therefore, we propose an iterative debiasing method wherein we train subsequent non-linear classifiers C_i to detect gender from $w^{(a)}$, and then adjust the autoencoder to fool all the classifiers. Thus, step by step, all gender information is incrementally eliminated from $w^{(a)}$ until no classifier can recover it. The iterative adversarial process of disentangling gender from general semantics is formalized in Algorithm 1 and illustrated in Figure 1

It should be noted that gender information is not lost entirely. While $w^{(a)}$ would be free of it after enough iterations, the autoencoder is trained to steer gender signals into $w^{(g)}$ sub-vectors, such that the decoder would be able to correctly reconstruct the original word. To do that, we first categorize the training vocabulary

[3] In practice we set $g = 1$ and $d = a + g$

Algorithm 1: Iterative Adversarial Removal of Gender in Word Embeddings

Input 1 : (X,Y): a training set of word vectors and their gender labels
Input 2 : n: number of iterations
Result: An encoder model E
$E, D \leftarrow pretrain\_autoencoder(X)$;
$classifiers \leftarrow [] ;$
for $i \leftarrow 1 \quad to \quad n$ **do**
  $X^{(a)}, X^{(g)} \leftarrow E(X) ;$
  $C_i \leftarrow train\_classifier(X^{(a)}, Y) ;$
  $classifiers.append(C_i) ;$
  $E, D \leftarrow train\_autoencoder(X, Y, classifiers) ;$
end
Return E ;

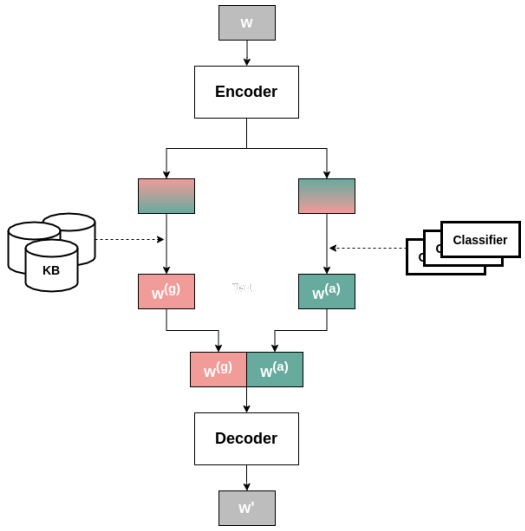


Figure 1: Iterative adversarial disentanglement of gender from general semantics in pretrained word embeddings

into three non-overlapping subsets: male-definition Ω^M, female-definition Ω^F and gender-neutral Ω^N. The component $w^{(g)}$ of every word embedding should verify the following:

- for $w_i \in \Omega^M, w_i^{(g)} \approx 1$
- for $w_i \in \Omega^F, w_i^{(g)} \approx -1$
- for $w_i \in \Omega^N, w_i^{(g)} \approx 0$

Finally, debiasing becomes straightforward. Given that every word embedding is disentangled nicely into coherent gender and semantics sub-vectors, we conduct debiasing by setting $w^{(g)}$ of a supposedly gender-free word embedding to 0, thus eliminating gender bias completely from such words. Then, we use the decoder to return to the original embedding space. We do not change $w^{(g)}$ of words that are inherently gendered such as *beard* or *pregnant* in order not to lose useful gender information. We remind that our goal is to remove gender bias from words that should be free of it, and not eliminate gender from all words.

To select the set of words to be debiased, we extract gender identity of words from existing lexical knowledge bases. Specifically, we use dictionaries and follow Kumar et al. [17] gender assumption. Namely, we define a word w to be gender-specific if there exists a dictionary d such that its definition corresponding to w ($d[w]$) contains a gender-specific reference $s \in \Omega^M \cup \Omega^F$ such as *man*, *he* or *mother*. We believe that the existence of these references in the dictionary definition of a word is a telltale sign that the word is inherently gendered. We only debias words whose definitions lack such references. In the following, we provide mathematical details about our debiasing method.

3.2 Formulation

The classifiers are trained using weighted cross entropy loss. However, the minimization objective of the autoencoder training procedure has three components:

$$\mathcal{L} = \lambda_R \mathcal{L}_R + \lambda_G \mathcal{L}_G + \lambda_A \mathcal{L}_A \tag{1}$$

Here, λ_R, λ_G and λ_A are non-negative hyperparameters that determine the relative importance of each component in Equation 1 compared to the others.

In the following, let us denote Ω as the set of word vectors available at training ($\Omega = \Omega^M \cup \Omega^F \cup \Omega^N$), E is the encoder model and D the decoder model. For every word w in Ω, the encoder E splits the latent representation in two sub-vectors as mentioned above.

$$w^{(a)}, w^{(g)} = E(w) \tag{2}$$

The first component $\mathcal{L}_R$ in Equation 1 is the standard reconstruction loss of autoencoders, which preserves the semantic and analogical properties of word vectors.

$$\mathcal{L}_R = \sum_{w \in \Omega} ||D(w^{(a)}, w^{(g)}) - w||_2^2 \tag{3}$$

This is important since debiasing is likely to alter the new embeddings. $\mathcal{L}_R$ prevents the autoencoder from changing the latent structure too much because it forces the decoder to still be able to reconstruct the orginal embedding given the two sub-vectors of the latent space.

Debiasing is ensured by the following two terms in Equation 1. $\mathcal{L}_G$ encodes gender information in $w^{(g)}$. Masculine words are encouraged to store a value of 1 in their gender sub-vectors, while feminine words that of -1. $\mathcal{L}_G^N$ forces gender-neutral words to have no gender information.

$$\mathcal{L}_G = \mathcal{L}_G^M + \mathcal{L}_G^F + \mathcal{L}_G^N \tag{4}$$

$$\mathcal{L}_G^M = \sum_{w \in \Omega^M} ||w^{(g)} - 1||_2^2 \tag{5}$$

$$\mathcal{L}_G^F = \sum_{w \in \Omega^F} ||w^{(g)} + 1||_2^2 \tag{6}$$

$$\mathcal{L}_G^N = \sum_{w \in \Omega^N} ||w^{(g)}||_2^2 \tag{7}$$

Finally, the last term $\mathcal{L}_A$ in the minimization objective protects $w^{(a)}$ from any gender influence in an adversarial fashion. We minimize the Kullback–Leibler divergence between the softmax logits produced by the set of all classifiers that have been trained before timestep i, and a discrete uniform distribution with three values (male, female and neutral). The rationale is to make the classifiers clueless about the gender identity of words encoded in $w^{(a)}$ by making them unsure about whether to classify inputs as male, female, or neutral; hence a uniform distribution of classifiers' predictions across these three classes.

$$\mathcal{L}_A = D_{KL}(\sum_{j=1}^{i} Softmax(C_j(w^{(a)})) \quad || \quad u) \tag{8}$$

where C_j is a trained classifier at iteration j ($j <= i$), $Softmax(.)$ is a function that gives the softmax logits of the classifier's prediction, and $u \sim \mathcal{U}(3)$ is a 3-class uniform distribution. Therefore, the autoencoder learns a new representation for $w^{(a)}$ such that all gender classifiers trained thus far fail to recognize the gender identity of words.

4 EXPERIMENTS

4.1 Implementation Details

4.1.1 Training data and models. In our work, both the encoder E, the decoder D and the adversarial classifiers C_i are implemented as feed-forward neural networks with two hidden layers. The activation functions we used are the hyperbolic tangent (tanh) for the autoencoder, and Rectified Linear Unit (ReLU) for the classifiers.

We picked the training data from previous work. We use the feminine and masculine words compiled by Zhao et al. [40] comprising of 223 words each. As for the gender-neutral wordlist, we utilize that created by Kaneko and Bollegala [14] consisting of 1031 words manually verified for their gender-neutrality. Finally, we use GloVe embeddings [29] with 300 dimensions and 322636 unique tokens, pretrained on 2017 January dump of English Wikipedia.

4.1.2 Training Details. We used Adam optimizer with a learning rate of $1e^{-6}$ for the autoencoder and $1e^{-5}$ for the classifiers. To overcome overfitting, we used dropout with a ratio of 20% neurons to be deactivated. We conducted the debiasing procedure described in Algorithm 1 for 30 iterations, and we selected the training coefficients as follows: $\lambda_R = 1$, $\lambda_G = 0.9$, $\lambda_A = 0.9$ before normalization. We conducted hyperparameter search manually.

4.2 Baselines

In our experiments, we compare our method against several baselines from literature.

4.2.1 GloVe: is a word embedding model pre-trained on 2017 January dump of English Wikipedia. This represents the non-debiased baseline of word embeddings.

4.2.2 Hard-GloVe: The authors of Hard-Debias [2] evaluated their method on word2vec [26]. We use their implementation[4], and apply it on GloVe embeddings for meaningful comparisons.

4.2.3 GP-GloVe: We use the gender-preserving debiased version of GloVe using an autoencoder, proposed and released[5] by [14].

4.2.4 RAN-GloVe. : This method debiased the original GloVe embeddings by altering the vector space with Repulsion and Attraction mechanisms. The authors [17] released their embeddings[6], that we use off-the-shelf.

4.2.5 ADV-GloVe: We apply the proposed debiasing methodology presented in this paper to reduce gender bias from the original GloVe embeddings. We call it ADV-GloVe owing to the use of adversarial training.

We purposefully exclude GN-GloVe [40] from this discussion since it incurs greater costs by retraining word embeddings from scratch. On the other hand, all baselines presented above have similar costs to our method (ADV-GloVe) in that they are all based on finetuning. Therefore, comparisons against these baselines are meaningful and fair.

4.3 Debiasing Performance

To evaluate the extent of gender bias in word embeddings, we use the popular dataset *SemBias* created by Zhao et al. [40]. Each instance in *SemBias* contains four word pairs: a gender-definition word pair (**Definition**; e.g., "gentleman - lady"), a gender-stereotype word pair (**Stereotype**; e.g., "doctor - nurse"); the two other pairs consist of words similar in meaning but irrelevant to gender (**None**; e.g., "cat - dog", or "flour - sugar"). *SemBias* contains 440 instances which have been constructed by the Cartesian product of 22 gender-definition word pairs and 20 gender-stereotype word pairs. Among the gender-definition word pairs, Zhao et al. [40] excluded 2 of them from their training procedure, and used them to test the generalization properties of their model. In the same spirit, we use *SemBias-Subset* which contains 40 instances associated with the excluded 2 pairs. In each instance, we look for the word pair whose relation is most similar to that of *he* and *she*. Here, word relations are defined by vector differences. Specifically, $\overrightarrow{he} - \overrightarrow{she}$ defines a gender relation since the only difference between *he* and *she* is gender. Ideally, a non-biased embedding model would find that the vector difference of the gender-definition pair is always the most similar to (he, she) among the four pairs in each instance, meaning that the gender-definition pair is the one that encodes gender more than the other pairs. To measure similarity between (he, she) and a pair (a, b) from *SemBias*, we use cosine similarity between the vectors $\overrightarrow{he} - \overrightarrow{she}$ and $\overrightarrow{a} - \overrightarrow{b}$ utilizing the embedding model under evaluation. We select the class (**Definition**, **Stereotype** or **None**) of the pair having the highest cosine similarity with the gender direction in each instance as the predicted answer. Table 1 reports the percentages where an instance in *SemBias* (or *SemBias-Subset*) is correctly classified as **Definition**, **Stereotype** or **None**. As mentioned above, an ideal embedding model maximizes the accuracy of **Definition** while it minimizes that of the other classes.

Table 1 confirms that the original GloVe embeddings are indeed biased with respect to gender since they have the lowest accuracies in **Definition** and highest accuracies in **Stereotype** and **None**. As can be seen, all baselines from the literature manage to reduce

[4] https://github.com/tolga-b/debiaswe

[5] https://github.com/kanekomasahiro/gp_debias

[6] https://github.com/TimeTraveller-San/RAN-Debias

Table 1: Comparison of gender relational analogy on SemBias dataset. ↑ (↓) indicate that higher (lower) values are better.

Embeddings	*SemBias*			*SemBias-Subset*		
	Definition ↑	Stereotype ↓	None ↓	Definition ↑	Stereotype ↓	None ↓
GloVe	80.22	10.91	8.86	57.5	20.0	22.5
Hard-GloVe	76.36	15.91	7.73	2.5	62.5	35.0
GP-GloVe	84.32	7.95	7.73	65.0	15.0	20.0
RAN-GloVe	92.73	1.14	6.14	97.5	**0.0**	2.5
ADV-GloVe	**95.45**	**0.91**	**3.64**	**100.0**	**0.0**	**0.0**

Table 2: Spearman correlations between cosine similarity and human ratings.

Embeddings	RG	WS	MTurk	MEN	SimLex
GloVe	75.30	61.12	64.87	72.99	34.72
Hard-GloVe	**76.35**	61.13	65.05	72.82	34.99
GP-GloVe	75.36	59.01	63.91	70.82	33.88
RAN-GloVe	76.22	60.92	64.31	72.81	34.22
ADV-GloVe	75.75	**65.68**	**65.17**	**73.14**	**36.73**

gender bias in GloVe embeddings[7]. Interestingly, our method outperforms the baselines in both versions of SemBias. We believe that these excellent results are owed to the fact that we remove non-linear bias (through the use of non-linear adversaries) whereas most previous works we compare our method against stop at linear bias removal (due to linear projections on the gender direction).

4.4 Semantic Similarity Test

While debiasing word embeddings, it is important to only remove information related to inappropriate and biased gender connotations, and preserve the general semantics of the embedding model in order for it to be usable in downstream NLP tasks. In this experiment, we evaluate how much debiasing alters the semantic space. Following previous work [2, 14, 17, 40], we define semantic accuracy as Spearman's correlation between the cosine similarity of a pair of words with its human-annotated rating. The higher the correlation, the better the underlying embedding model is at preserving semantic properties. We conduct this experiment with five similarity benchmarks: Rubenstein-Goodenough dataset (**RG**) [31], Word Similarity 353 dataset (**WS**) [10], **MTurk** [12], **MEN** [4] and **SimLex** dataset [13]. It is important to note that in this experiment, we do not aim to score state-of-the-art semantic accuracies, but we are interested in quantifying semantic loss after debiasing. Table 2 shows the results.

Compared with others, our method achieves high correlation with ground-truth similarity, indicating that we introduce minimal semantic disturbance. Indeed, as with the previous experiment, ADV-GloVe is the best in most similarity tasks. What's more, we note that debiasing improves the semantic offset of the original GloVe embeddings by a relative margin of up to 7.46%.

[7] Apart from Hard-GloVe which was originally tested on Word2vec

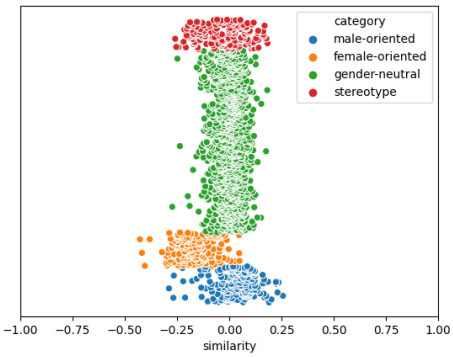
Table 3: F1 score (%) on the coreference task

Embeddings	**OntoNotes**	**PRO**	**ANTI**	**Avg**	**|Diff|**
GloVe	71.99	74.34	50.15	62.25	24.19
Hard-GloVe	71.90	75.03	52.46	63.75	22.57
GP-GloVe	71.67	75.90	51.06	63.48	24.84
RAN-GloVe	71.56	73.51	53.04	63.28	20.47
ADV-GloVe	71.70	72.82	52.82	62.82	**20.0**

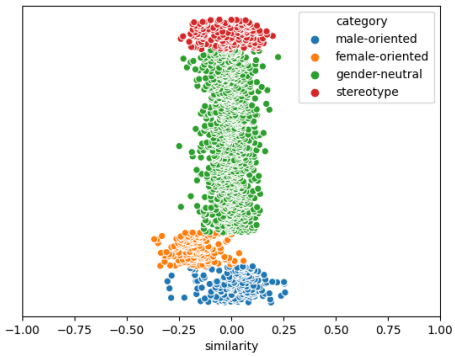
4.5 Co-reference Resolution Test

We investigate the performance of the newly constructed word vectors in their capacity to assist a co-reference resolution model without skewing it toward biased decisions. In this experiment, we use the model proposed by Lee et al. [20], which counts among the best co-reference resolution models in the literature. We train it using OntoNotes 5.0 dataset [37], and the embedding models presented in Section 4.2, one at a time. We keep the same hyperparameters as in the original paper, and train the co-reference model for 70k steps. To assess the degree of gender stereotype exhibited by the downstream co-reference system, we utilize WinoBias dataset [39], which comprises pro-stereotypical (PRO) and anti-stereotypical (ANTI) subsets. In the PRO subset, gender pronouns (he or she) refer to occupation terms in line with the same pronoun's gender. On the other hand, the gender connotations of occupation terms and pronouns are opposite in the ANTI subset. For example, consider the sentence: *"The physician hired the secretary because **[Blank]** was overwhelmed with clients"*. The blank is replaced by *he* in PRO subset because the occupation of *physician* is stereotyped to be rather masculine than feminine. Likewise, it is replaced by *she* in ANTI subset. In this sentence, the blank refers to the physician no matter the gender of the pronoun that would replace it. Following the same example, a biased co-reference model might be enticed to refer the pronoun *she* (in ANTI subset) to the occupation *secretary* rather than *physician* because of gender bias influence. Consequently, we would expect a biased co-reference model to have a considerably harder time to predict correct answers for ANTI than for PRO. Table 3 reports the F1 scores of the resulting models trained with the respective word embeddings on OntoNotes test set, PRO and ANTI subsets. We also report the average F1 score, and their difference ($|Diff|$) between PRO and ANTI since it works as a direct proxy for quantifying gender bias. A smaller Diff value indicates less bias.

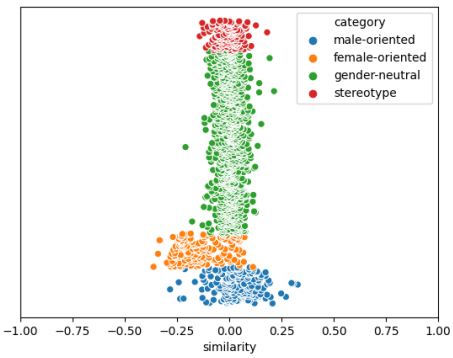
Results show that our method reduces gender bias when the resulting embeddings are applied in a co-reference resolution context (4.19% decrease in the difference metric). We also show that



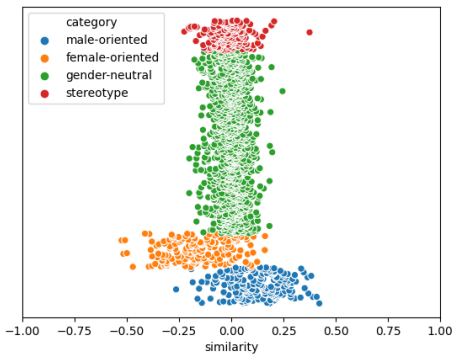
(a) GloVe



(b) GP-GloVe



(c) RAN-GloVe



(d) ADV-GloVe

Figure 2: Cosine Similarity between gender-definition, gender-neutral and gender-stereotype words and the gender direction defined by $\overrightarrow{he} - \overrightarrow{she}$. X-axis: cosine similarities (positive values lean to masculinity while negative values lean toward femininity). Y-axis: Random values to separate the datapoints in the visualizations.

the usability of the embeddings after debiasing is not hindered as F1 scores on OntoNotes test set has decreased only slightly from the undisturbed embeddings. Finally, we want to emphasize that in spite of this and the previous experiments which demonstrated that the new embeddings are certainly *less* biased with regard to gender, we are still unable to eliminate all unfair gender cues completely (the $|Diff|$ scores are still significant[8]). More effort and investigation are still called for in this area.

4.6 Visualization of Debiased Word Embeddings

In this experiment, we aim to visualize the effect of debiasing. To do that, we compute cosine similarity of every word with the gender direction defined by the vector difference $\overrightarrow{he} - \overrightarrow{she}$. Intuitively, a high positive similarity score indicates a strong inclination of the word under evaluation to lean toward the masculine side, whereas negative scores suggest feminine interpretations. A cosine similarity centered around zero implies perpendicularity of the word vector in question with the gender direction, hence neutrality of gender. We collect four sets of words from Kaneko and Bollegala [14]: male-oriented, female-oriented, gender-neutral and gender-stereotyped words. The latter comprises words that should be free of gender influence but have been established as stereotyped because they pick a side in the gender spectrum (e.g., occupation words which should be neutral but are substantially associated with one gender more than the other: doctor → male, nurse → female). We plot the cosine similarities of these four sets of words with the gender direction in Figure 2, where the x-axis represents the similarities, and the y-axis random values to separate the words vertically.

We see that in the original GloVe vectors, the spread of gender-stereotype words is wider than that of gender-neutral, which means that gender-stereotype words still encode gender cues. Besides, male-oriented and female-oriented words are, to some extent, clustered around the middle, indicating a poor representation of gender in GloVe embeddings. We observe that debiasing baselines also suffer from these two limitations, apart from RAN-GloVe which brings the spread of gender-neutral and gender-stereotype words

[8]These high $|Diff|$ scores can also be due to biased training data, not only biased word embeddings

to the same width, but still struggles to clearly differentiate between male-oriented and female-oriented words. In contrast, the aforementioned issues are solved by ADV-GloVe which reduces the spread of gender-neutral and gender-stereotype words to the same width (i.e., removing unfair and illegitimate gender cues). Also, ADV-GloVe pushes each gendered cluster to its expected whereabouts in Figure 2: male words share high positive cosine similarities with the gender direction, and female words take on negative similarities (i.e., retaining and emphasizing meaningful gender information in word embeddings).

5 CONCLUSION

Despite recent findings, we showed that adversarial training can be effective in removing sensitive information from neural representations if used in an iterative way. We applied our method in the context of reducing unfair and illegitimate gender bias from word embeddings while we retain meaningful gender information in inherently gendered words, all with minimal semantic offset. Experimental results demonstrate that our post-processing method outperforms existing approaches. As future work, we plan to extend our studies to non-binary gender, and other demographic attributes such as race, religion or ethnicity. Finally, we caution against trusting debiasing methods for completely mitigating gender bias in word embeddings. Although the bias under study is certainly reduced, chances are it is still lurking in shapes and forms that our experimental lenses failed to detect.

REFERENCES

[1] Mohsen Abbasi, Sorelle A Friedler, Carlos Scheidegger, and Suresh Venkatasubramanian. 2019. Fairness in representation: quantifying stereotyping as a representational harm. In *Proceedings of the 2019 SIAM International Conference on Data Mining*. SIAM, 801–809.

[2] Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *arXiv preprint arXiv:1607.06520* (2016).

[3] Marc-Etienne Brunet, Colleen Alkalay-Houlihan, Ashton Anderson, and Richard Zemel. 2019. Understanding the origins of bias in word embeddings. In *International Conference on Machine Learning*. PMLR, 803–811.

[4] Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 136–145.

[5] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science* 356, 6334 (2017), 183–186.

[6] R Dennis Cook and Sanford Weisberg. 1980. Characterizations of an empirical influence function for detecting influential cases in regression. *Technometrics* 22, 4 (1980), 495–508.

[7] Sunipa Dev, Tao Li, Jeff M Phillips, and Vivek Srikumar. 2020. On measuring and mitigating biased inferences of word embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 7659–7666.

[8] Sunipa Dev, Tao Li, Jeff M Phillips, and Vivek Srikumar. 2020. OSCaR: Orthogonal Subspace Correction and Rectification of Biases in Word Embeddings. *arXiv preprint arXiv:2007.00049* (2020).

[9] Yanai Elazar and Yoav Goldberg. 2018. Adversarial removal of demographic attributes from text data. *arXiv preprint arXiv:1808.06640* (2018).

[10] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. 2001. Placing search in context: The concept revisited. In *Proceedings of the 10th international conference on World Wide Web*. 406–414.

[11] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[12] Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 1406–1414.

[13] Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics* 41, 4 (2015), 665–695.

[14] Masahiro Kaneko and Danushka Bollegala. 2019. Gender-preserving debiasing for pre-trained word embeddings. *arXiv preprint arXiv:1906.00742* (2019).

[15] Masahiro Kaneko and Danushka Bollegala. 2021. Dictionary-based debiasing of pre-trained word embeddings. *arXiv preprint arXiv:2101.09525* (2021).

[16] Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*. PMLR, 1885–1894.

[17] Vaibhav Kumar, Tenzin Singhay Bhotia, and Tanmoy Chakraborty. 2020. Nurse is closer to woman than surgeon? mitigating gender-biased proximities in word embeddings. *Transactions of the Association for Computational Linguistics* 8 (2020), 486–503.

[18] Keita Kurita, Nidhi Vyas, Ayush Pareek, Alan W Black, and Yulia Tsvetkov. 2019. Quantifying social biases in contextual word representations. In *1st ACL Workshop on Gender Bias for Natural Language Processing*.

[19] Brian Larson. 2017. Gender as a Variable in Natural-Language Processing: Ethical Considerations. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*. 1–11.

[20] Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. *arXiv preprint arXiv:1804.05392* (2018).

[21] Tao Li, Tushar Khot, Daniel Khashabi, Ashish Sabharwal, and Vivek Srikumar. 2020. UNQOVERing stereotyping biases via underspecified questions. *arXiv preprint arXiv:2010.02428* (2020).

[22] Yitong Li, Timothy Baldwin, and Trevor Cohn. 2018. Towards robust and privacy-preserving text representations. *arXiv preprint arXiv:1805.06093* (2018).

[23] Haochen Liu, Wentao Wang, Yiqi Wang, Hui Liu, Zitao Liu, and Jiliang Tang. 2020. Mitigating Gender Bias for Neural Dialogue Generation with Adversarial Learning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 893–903.

[24] Thomas Manzini, Yao Chong Lim, Yulia Tsvetkov, and Alan W Black. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. *arXiv preprint arXiv:1904.04047* (2019).

[25] Chandler May, Alex Wang, Shikha Bordia, Samuel R Bowman, and Rachel Rudinger. 2019. On Measuring Social Biases in Sentence Encoders. In *NAACL-HLT (1)*.

[26] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781* (2013).

[27] Moin Nadeem, Anna Bethke, and Siva Reddy. 2020. Stereoset: Measuring stereotypical bias in pretrained language models. *arXiv preprint arXiv:2004.09456* (2020).

[28] Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel 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)*. 1953–1967.

[29] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 1532–1543.

[30] Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null it out: Guarding protected attributes by iterative nullspace projection. *arXiv preprint arXiv:2004.07667* (2020).

[31] Herbert Rubenstein and John B Goodenough. 1965. Contextual correlates of synonymy. *Commun. ACM* 8, 10 (1965), 627–633.

[32] Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. *arXiv preprint arXiv:1804.09301* (2018).

[33] Jürgen Schmidhuber. 2015. Deep learning in neural networks: An overview. *Neural networks* 61 (2015), 85–117.

[34] Emily Sheng, Kai-Wei Chang, Prem Natarajan, and Nanyun Peng. 2020. Towards Controllable Biases in Language Generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*. 3239–3254.

[35] Gabriel Stanovsky, Noah A Smith, and Luke Zettlemoyer. 2019. Evaluating gender bias in machine translation. *arXiv preprint arXiv:1906.00591* (2019).

[36] Tianlu Wang, Xi Victoria Lin, Nazneen Fatema Rajani, Bryan McCann, Vicente Ordonez, and Caiming Xiong. 2020. Double-hard debias: Tailoring word embeddings for gender bias mitigation. *arXiv preprint arXiv:2005.00965* (2020).

[37] Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, et al. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA* 23 (2013).

[38] Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. 2017. Controllable invariance through adversarial feature learning. *arXiv preprint arXiv:1705.11122* (2017).

[39] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. *arXiv preprint arXiv:1804.06876* (2018).

[40] Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018. Learning gender-neutral word embeddings. *arXiv preprint arXiv:1809.01496* (2018).