Dictionary-based Debiasing of Pre-trained Word Embeddings

Masahiro Kaneko
Tokyo Metropolitan University
kaneko-masahiro@ed.tmu.ac.jp

Danushka Bollegala*
University of Liverpool, Amazon
danushka@liverpool.ac.uk

Abstract

Word embeddings trained on large corpora have shown to encode high levels of unfair discriminatory gender, racial, religious and ethnic biases. In contrast, human-written dictionaries describe the meanings of words in a concise, objective and an unbiased manner. We propose a method for debiasing pre-trained word embeddings using dictionaries, without requiring access to the original training resources or any knowledge regarding the word embedding algorithms used. Unlike prior work, our proposed method does not require the types of biases to be pre-defined in the form of word lists, and learns the constraints that must be satisfied by unbiased word embeddings automatically from dictionary definitions of the words. Specifically, we learn an encoder to generate a debiased version of an input word embedding such that it (a) retains the semantics of the pre-trained word embeddings, (b) agrees with the unbiased definition of the word according to the dictionary, and (c) remains orthogonal to the vector space spanned by any biased basis vectors in the pre-trained word embedding space. Experimental results on standard benchmark datasets show that the proposed method can accurately remove unfair biases encoded in pre-trained word embeddings, while preserving useful semantics.

1 Introduction

Despite pre-trained word embeddings are useful due to their low dimensionality, memory and compute efficiency, they have shown to encode not only the semantics of words but also unfair discriminatory biases such as gender, racial or religious biases (Bolukbasi et al., 2016; Zhao et al., 2018a; Rudinger et al., 2018; Zhao et al., 2018b; Elazar and Goldberg, 2018; Kaneko and Bollegala, 2019). On the other hand, human-written dictionaries act as an impartial, objective and unbiased source of word meaning. Although methods that learn word embeddings by purely using dictionaries have been proposed (Tissier et al., 2017), they have coverage and data sparseness related issues because pre-compiled dictionaries do not capture the meanings of neologisms or provide numerous contexts as in a corpus. Consequently, prior work has shown that word embeddings learnt from large text corpora to outperform those created from dictionaries in downstream NLP tasks (Alsuhaibani et al., 2019; Bollegala et al., 2016).

We must overcome several challenges when using dictionaries to debias pre-trained word embeddings. First, not all words in the embeddings will appear in the given dictionary. Dictionaries often have limited coverage and will not cover neologisms, orthographic variants of words etc. that are likely to appear in large corpora. A lexicalised debiasing method would generalise poorly to the words not in the dictionary. Second, it is not known apriori what biases are hidden inside a set of pre-trained word embedding vectors. Depending on the source of documents used for training the embeddings, different types of biases will be learnt and amplified by different word embedding learning algorithms to different degrees (Zhao et al., 2017).

Prior work on debiasing required that the biases to be pre-defined (Kaneko and Bollegala, 2019). For example, Hard-Debias (HD; Bolukbasi et al., 2016) and Gender Neutral Glove (GN-GloVe; Zhao et al., 2018b) require lists of *male* and *female* pronouns for defining the *gender* direction. However, gender bias is only one of the many biases that exist in pre-trained word embeddings. It is inconvenient to prepare lists of words covering all different types of biases we must remove from pre-trained word embeddings. Moreover, such pre-

*Danushka Bollegala holds concurrent appointments as a Professor at University of Liverpool and as an Amazon Scholar. This paper describes work performed at the University of Liverpool and is not associated with Amazon.

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 212–223
April 19 - 23, 2021.

compiled word lists are likely to be incomplete and inadequately cover some biases. Indeed, Gonen and Goldberg (2019) showed empirical evidence that such debiasing methods do not remove all discriminative biases from word embeddings. Unfair biases have adversely affected several NLP tasks such as machine translation (Vanmassenhove et al., 2018) and language generation (Sheng et al., 2019). Racial biases have also been shown to affect criminal prosecutions (Manzini et al., 2019) and career adverts (Lambrecht and Tucker, 2016). These findings show the difficulty of defining different biases using pre-compiled lists of words, which is a requirement in previously proposed debiasing methods for static word embeddings.

We propose a method that uses a dictionary as a source of bias-free definitions of words for debiasing pre-trained word embeddings[1]. Specifically, we learn an encoder that filters-out biases from the input embeddings. The debiased embeddings are required to simultaneously satisfy three criteria: (a) must preserve all non-discriminatory information in the pre-trained embeddings (*semantic preservation*), (b) must be similar to the dictionary definition of the words (*dictionary agreement*), and (c) must be orthogonal to the subspace spanned by the basis vectors in the pre-trained word embedding space that corresponds to discriminatory biases (*bias orthogonality*). We implement the semantic preservation and dictionary agreement using two decoders, whereas the bias orthogonality is enforced by a parameter-free projection. The debiasing encoder and the decoders are learnt end-to-end by a joint optimisation method. Our proposed method is agnostic to the details of the algorithms used to learn the input word embeddings. Moreover, unlike counterfactual data augmentation methods for debiasing (Zmigrod et al., 2019; Hall Maudslay et al., 2019), we do *not* require access to the original training resources used for learning the input word embeddings.

Our proposed method overcomes the above-described challenges as follows. First, instead of learning a lexicalised debiasing model, we operate on the word embedding space when learning the encoder. Therefore, we can use the words that are in the intersection of the vocabularies of the pre-trained word embeddings and the dictionary to learn the encoder, enabling us to generalise to the words not in the dictionary. Second, we do *not* require pre-compiled word lists specifying the biases. The dictionary acts as a clean, unbiased source of word meaning that can be considered as *positive* examples of debiased meanings. In contrast to the existing debiasing methods that require us to predefine *what to remove*, the proposed method can be seen as using the dictionary as a guideline for *what to retain* during debiasing.

We evaluate the proposed method using four standard benchmark datasets for evaluating the biases in word embeddings: Word Embedding Association Test (WEAT; Caliskan et al., 2017), Word Association Test (WAT; Du et al., 2019), SemBias (Zhao et al., 2018b) and WinoBias (Zhao et al., 2018a). Our experimental results show that the proposed debiasing method accurately removes unfair biases from three widely used pre-trained embeddings: Word2Vec (Mikolov et al., 2013b), GloVe (Pennington et al., 2014) and fastText (Bojanowski et al., 2017). Moreover, our evaluations on semantic similarity and word analogy benchmarks show that the proposed debiasing method preserves useful semantic information in word embeddings, while removing unfair biases.

2 Related Work

Dictionaries have been popularly used for learning word embeddings (Budanitsky and Hirst, 2006, 2001; Jiang and Conrath, 1997). Methods that use both dictionaries (or lexicons) and corpora to jointly learn word embeddings (Tissier et al., 2017; Alsuhaibani et al., 2019; Bollegala et al., 2016) or post-process (Glavaš and Vulić, 2018; Faruqui et al., 2015) have also been proposed. However, learning embeddings from dictionaries alone results in coverage and data sparseness issues (Bollegala et al., 2016) and does not guarantee bias-free embeddings (Lauscher and Glavas, 2019). To the best of our knowledge, we are the first to use dictionaries for debiasing pre-trained word embeddings.

Bolukbasi et al. (2016) proposed a post-processing approach that projects gender-neutral words into a subspace, which is orthogonal to the gender dimension defined by a list of gender-definitional words. They refer to words associated with gender (e.g., *she*, *actor*) as gender-definitional words, and the remainder gender-neutral. They proposed a *hard-debiasing* method where the gender direction is computed as the vector difference between the embeddings of the correspond-

[1]Code and debiased embeddings: https://github.com/kanekomasahiro/dict-debias

ing gender-definitional words, and a *soft-debiasing* method, which balances the objective of preserving the inner-products between the original word embeddings, while projecting the word embeddings into a subspace orthogonal to the gender definitional words. Both hard and soft debiasing methods ignore gender-definitional words during the subsequent debiasing process, and focus only on words that are *not* predicted as gender-definitional by the classifier. Therefore, if the classifier erroneously predicts a stereotypical word as a gender-definitional word, it would not get debiased.

Zhao et al. (2018b) modified the GloVe (Pennington et al., 2014) objective to learn gender-neutral word embeddings (GN-GloVe) from a given corpus. They maximise the squared ℓ_2 distance between gender-related sub-vectors, while simultaneously minimising the GloVe objective. Unlike, the above-mentioned methods, Kaneko and Bollegala (2019) proposed a post-processing method to preserve gender-related information with autoencoder (Kaneko and Bollegala, 2020), while removing discriminatory biases from stereotypical cases (GP-GloVe). However, all prior debiasing methods require us to pre-define the biases in the form of explicit word lists containing gender and stereotypical word associations. In contrast we use dictionaries as a source of bias-free semantic definitions of words and do not require pre-defining the biases to be removed. Although we focus on static word embeddings in this paper, unfair biases have been found in contextualised word embeddings as well (Zhao et al., 2019; Vig, 2019; Bordia and Bowman, 2019; May et al., 2019).

Adversarial learning methods (Xie et al., 2017; Elazar and Goldberg, 2018; Li et al., 2018) for debiasing first encode the inputs and then two classifiers are jointly trained – one predicting the target task (for which we must ensure high prediction accuracy) and the other protected attributes (that must not be easily predictable). However, Elazar and Goldberg (2018) showed that although it is possible to obtain chance-level development-set accuracy for the protected attributes during training, a post-hoc classifier trained on the encoded inputs can still manage to reach substantially high accuracies for the protected attributes. They conclude that adversarial learning alone does not guarantee invariant representations for the protected attributes. Ravfogel et al. (2020) found that iteratively projecting word embeddings to the null space of the gender direction to further improve the debiasing performance.

To evaluate biases, Caliskan et al. (2017) proposed the Word Embedding Association Test (WEAT) inspired by the Implicit Association Test (IAT; Greenwald et al., 1998). Ethayarajh et al. (2019) showed that WEAT to be systematically overestimating biases and proposed a correction. The ability to correctly answer gender-related word analogies (Zhao et al., 2018b) and resolve gender-related coreferences (Zhao et al., 2018a; Rudinger et al., 2018) have been used as extrinsic tasks for evaluating the bias in word embeddings. We describe these evaluation benchmarks later in § 4.3.

3 Dictionary-based Debiasing

Let us denote the n-dimensional pre-trained word embedding of a word w by $\boldsymbol{w} \in \mathbb{R}^n$ trained on some resource $\mathcal{C}$ such as a text corpus. Moreover, let us assume that we are given a dictionary $\mathcal{D}$ containing the definition, $s(w)$ of w. If the pre-trained embeddings distinguish among the different senses of w, then we can use the gloss for the corresponding sense of w in the dictionary as $s(w)$. However, the majority of word embedding learning methods do not produce sense-specific word embeddings. In this case, we can either use all glosses for w in $\mathcal{D}$ by concatenating or select the gloss for the dominant (most frequent) sense of w[2]. Without any loss of generality, in the remainder of this paper, we will use $s(w)$ to collectively denote a gloss selected by any one of the above-mentioned criteria with or without considering the word senses (in § 5.3, we evaluate the effect of using all vs. dominant gloss).

Next, we define the objective functions optimised by the proposed method for the purpose of learning unbiased word embeddings. Given, $\boldsymbol{w}$, we model the debiasing process as the task of learning an encoder, $E(\boldsymbol{w}; \boldsymbol{\theta}_e)$ that returns an $m(\leq n)$-dimensional debiased version of $\boldsymbol{w}$. In the case where we would like to preserve the dimensionality of the input embeddings, we can set $m = n$, or $m < n$ to further compress the debiased embeddings.

Because the pre-trained embeddings encode rich semantic information from a large text corpora, often far exceeding the meanings covered in the

[2]Prior work on debiasing static word embeddings do not use contextual information that is required for determining word senses. Therefore, for comparability reasons we do neither.

dictionary, we must preserve this semantic information as much as possible during the debiasing process. We refer to this constraint as *semantic preservation*. Semantic preservation is likely to lead to good performance in downstream NLP applications that use pre-trained word embeddings. For this purpose, we decode the encoded version of $\boldsymbol{w}$ using a decoder, D_c, parametrised by $\boldsymbol{\theta}_c$ and define J_c to be the reconstruction loss given by (1).

$$J_c(w) = ||\boldsymbol{w} - D_c(E(\boldsymbol{w}; \boldsymbol{\theta}_e); \boldsymbol{\theta}_c)||_2^2 \quad (1)$$

Following our assumption that the dictionary definition, $s(w)$, of w is a concise and unbiased description of the meaning of w, we would like to ensure that the encoded version of $\boldsymbol{w}$ is similar to $s(w)$. We refer to this constraint as *dictionary agreement*. To formalise dictionary agreement empirically, we first represent $s(w)$ by a sentence embedding vector $\boldsymbol{s}(w) \in \mathbb{R}^n$. Different sentence embedding methods can be used for this purpose such as convolutional neural networks (Kim, 2014), recurrent neural networks (Peters et al., 2018) or transformers (Devlin et al., 2019). For the simplicity, we use the smoothed inverse frequency (SIF; Arora et al., 2017) for creating $\boldsymbol{s}(w)$ in this paper. SIF computes the embedding of a sentence as the weighted average of the pre-trained word embeddings of the words in the sentence, where the weights are computed as the inverse unigram probability. Next, the first principal component vector of the sentence embeddings are removed. The dimensionality of the sentence embeddings created using SIF is equal to that of the pre-trained word embeddings used. Therefore, in our case we have both $\boldsymbol{w}, \boldsymbol{s}(w) \in \mathbb{R}^n$.

We decode the debiased embedding $E(\boldsymbol{w}; \boldsymbol{\theta}_e)$ of w using a decoder D_d, parametrised by $\boldsymbol{\theta}_d$ and compute the squared ℓ_2 distance between it and $\boldsymbol{s}(w)$ to define an objective J_d given by (2).

$$J_d(w) = ||\boldsymbol{s}(w) - D_d(E(\boldsymbol{w}; \boldsymbol{\theta}_e); \boldsymbol{\theta}_d)||_2^2 \quad (2)$$

Recalling that our goal is to remove unfair biases from pre-trained word embeddings and we assume dictionary definitions to be free of such biases, we define an objective function that explicitly models this requirement. We refer to this requirement as the *bias orthogonality* of the debiased embeddings. For this purpose, we first project the pre-trained word embedding $\boldsymbol{w}$ of a word w into a subspace that is orthogonal to the dictionary definition vector $\boldsymbol{s}(w)$. Let us denote this projection by $\phi(\boldsymbol{w}, \boldsymbol{s}(w)) \in \mathbb{R}^n$. We require that the debiased word embedding, $E(\boldsymbol{w}; \boldsymbol{\theta}_e)$, must be orthogonal to $\phi(\boldsymbol{w}, \boldsymbol{s}(w))$, and formalise this as the minimisation of the squared inner-product given in (3).

$$J_a(w) = \Big(E(\phi(\boldsymbol{w}, \boldsymbol{s}(w)); \boldsymbol{\theta}_e)^\top E(\boldsymbol{w}; \boldsymbol{\theta}_e)\Big)^2 \quad (3)$$

Note that because $\phi(\boldsymbol{w}, \boldsymbol{s}(w))$ lives in the space spanned by the original (prior to encoding) vector space, we must first encode it using E before considering the orthogonality requirement.

To derive $\phi(\boldsymbol{w}, \boldsymbol{s}(w))$, let us assume the n-dimensional basis vectors in the R^n vector space spanned by the pre-trained word embeddings to be $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_n$. Moreover, without loss of generality, let the subspace spanned by the subset of the first $k(< n)$ basis vectors $\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_k$ to be $\mathcal{B} \subseteq \mathbb{R}^n$. The projection $\boldsymbol{v}_\mathcal{B}$ of a vector $\boldsymbol{v} \in \mathbb{R}^n$ onto $\mathcal{B}$ can be expressed using the basis vectors as in (4).

$$\boldsymbol{v}_\mathcal{B} = \sum_{j=1}^{k} (\boldsymbol{v}^\top \boldsymbol{b}_j) \boldsymbol{b}_j \quad (4)$$

To show that $\boldsymbol{v} - \boldsymbol{v}_\mathcal{B}$ is orthogonal to $\boldsymbol{v}_\mathcal{B}$ for any $\boldsymbol{v} \in \mathcal{B}$, let us express $\boldsymbol{v} - \boldsymbol{v}_\mathcal{B}$ using the basis vectors as given in (5).

$$\begin{aligned} \boldsymbol{v} - \boldsymbol{v}_\mathcal{B} &= \sum_{i=1}^{n} (\boldsymbol{v}^\top \boldsymbol{b}_i) \boldsymbol{b}_i - \sum_{j=1}^{k} (\boldsymbol{v}^\top \boldsymbol{b}_j) \boldsymbol{b}_j \\ &= \sum_{i=k+1}^{n} (\boldsymbol{v}^\top \boldsymbol{b}_i) \boldsymbol{b}_i \end{aligned} \quad (5)$$

We see that there are no basis vectors in common between the summations in (4) and (5). Therefore, $\boldsymbol{v}_\mathcal{B}{}^\top (\boldsymbol{v} - \boldsymbol{v}_\mathcal{B}) = 0$ for $\forall \boldsymbol{v} \in \mathcal{B}$.

Considering that $\boldsymbol{s}(w)$ defines a direction that does not contain any unfair biases, we can compute the vector rejection of $\boldsymbol{w}$ on $\boldsymbol{s}(w)$ following this result. Specifically, we subtract the projection of $\boldsymbol{w}$ along the unit vector defining the direction of $\boldsymbol{s}(w)$ to compute ϕ as in (6).

$$\phi(\boldsymbol{w}, \boldsymbol{s}(w)) = \boldsymbol{w} - \boldsymbol{w}^\top \boldsymbol{s}(w) \frac{\boldsymbol{s}(w)}{||\boldsymbol{s}(w)||} \quad (6)$$

We consider the linearly-weighted sum of the above-defined three objective functions as the total objective function as given in (7).

$$J(w) = \alpha J_c(w) + \beta J_d(w) + \gamma J_a(w) \quad (7)$$

Here, $\alpha, \beta, \gamma \geq 0$ are scalar coefficients satisfying $\alpha + \beta + \gamma = 1$. Later, in § 4 we experimentally determine the values of α, β and γ using a development dataset.

4 Experiments

4.1 Word Embeddings

In our experiments, we use the following publicly available pre-trained word embeddings: **Word2Vec**[3] (300-dimensional embeddings for ca. 3M words learned from Google News corpus (Mikolov et al., 2013a)), **GloVe**[4] (300-dimensional embeddings for ca. 2.1M words learned from the Common Crawl (Pennington et al., 2014)), and **fastText**[5] (300-dimensional embeddings for ca. 1M words learned from Wikipedia 2017, UMBC webbase corpus and statmt.org news (Bojanowski et al., 2017)).

As the dictionary definitions, we used the glosses in the WordNet (Fellbaum, 1998), which has been popularly used to learn word embeddings in prior work (Tissier et al., 2017; Bosc and Vincent, 2018; Washio et al., 2019). However, we note that our proposed method does not depend on any WordNet-specific features, thus in principle can be applied to any dictionary containing definition sentences. Words that do not appear in the vocabulary of the pre-trained embeddings are ignored when computing $\boldsymbol{s}(w)$ for the headwords w in the dictionary. Therefore, if all the words in a dictionary definition are ignored, then the we remove the corresponding headwords from training. Consequently, we are left with 54,528, 64,779 and 58,015 words respectively for **Word2Vec**, **GloVe** and **fastText** embeddings in the training dataset. We randomly sampled 1,000 words from this dataset and held-out as a development set for the purpose of tuning various hyperparameters in the proposed method.

E, D_c and D_d are implemented as single-layer feed forward neural networks with a hyperbolic tangent activation at the outputs. It is known that pre-training is effective when using autoencoders E and D_c for debiasing (Kaneko and Bollegala, 2019). Therefore, we randomly select 5000 words from each pre-trained word embedding set and pre-train the autoencoders on those words with a mini-batch of size 512. In pre-training, the model with the lowest loss according to (1) in the development set for pre-traininng is selected.

[3] https://code.google.com/archive/p/word2vec/

[4] https://github.com/stanfordnlp/GloVe

[5] https://fasttext.cc/docs/en/english-vectors.html

4.2 Hyperparameters

During optimisation, we used dropout (Srivastava et al., 2014) with probability 0.05 to $\boldsymbol{w}$ and $E(w)$. We used Adam (Kingma and Ba, 2015) with initial learning rate set to 0.0002 as the optimiser to find the parameters $\boldsymbol{\theta}_e$, $\boldsymbol{\theta}_c$, and $\boldsymbol{\theta}_d$ and a mini-batch size of 4. The optimal values of all hyperparameters are found by minimising the total loss over the development dataset following a Monte-Carlo search. We found these optimal hyperparameter values of $\alpha = 0.99998$, $\beta = 0.00001$ and $\gamma = 0.00001$. Note that the scale of different losses are different and the absolute values of hyperparameters do *not* indicate the significance of a component loss. For example, if we rescale all losses to the same range then we have $L_c = 0.005\alpha$, $L_d = 0.269\beta$ and $L_a = 21.1999\gamma$. Therefore, debiasing (L_d) and orthogonalisation (L_a) contributions are significant.

We utilized a GeForce GTX 1080 Ti. The debiasing is completed in less than an hour because our method is only a fine-tuning technique. The parameter size of our debiasing model is 270,900.

4.3 Evaluation Datasets

We use the following datasets to evaluate the degree of the biases in word embeddings.

WEAT: Word Embedding Association Test (WEAT; Caliskan et al., 2017), quantifies various biases (e.g. gender, race and age) using semantic similarities between word embeddings. It compares two same size sets of *target* words $\mathcal{X}$ and $\mathcal{Y}$ (e.g. European and African names), with two sets of *attribute* words $\mathcal{A}$ and $\mathcal{B}$ (e.g. *pleasant* vs. *unpleasant*). The bias score, $s(\mathcal{X}, \mathcal{Y}, \mathcal{A}, \mathcal{B})$, for each target is calculated as follows:

$$s(\mathcal{X}, \mathcal{Y}, \mathcal{A}, \mathcal{B}) = \sum_{\boldsymbol{x} \in \mathcal{X}} k(\boldsymbol{x}, \mathcal{A}, \mathcal{B}) - \sum_{\boldsymbol{y} \in \mathcal{Y}} k(\boldsymbol{y}, \mathcal{A}, \mathcal{B}) \quad (8)$$

$$k(\boldsymbol{t}, \mathcal{A}, \mathcal{B}) = \text{mean}_{\boldsymbol{a} \in \mathcal{A}} f(\boldsymbol{t}, \boldsymbol{a}) - \text{mean}_{\boldsymbol{b} \in \mathcal{B}} f(\boldsymbol{t}, \boldsymbol{b}) \quad (9)$$

Here, f is the cosine similarity between the word embeddings. The one-sided p-value for the permutation test regarding $\mathcal{X}$ and $\mathcal{Y}$ is calculated as the probability of $s(\mathcal{X}_i, \mathcal{Y}_i, \mathcal{A}, \mathcal{B}) > s(\mathcal{X}, \mathcal{Y}, \mathcal{A}, \mathcal{B})$. The effect size is calculated as the normalised measure given by (10).

$$\frac{\text{mean}_{x \in \mathcal{X}} s(x, \mathcal{A}, \mathcal{B}) - \text{mean}_{y \in \mathcal{Y}} s(y, \mathcal{A}, \mathcal{B})}{\text{sd}_{t \in \mathcal{X} \cup \mathcal{Y}} s(t, \mathcal{A}, \mathcal{B})} \quad (10)$$

WAT: Word Association Test (WAT) is a method to measure gender bias over a large set of words (Du et al., 2019). It calculates the gender information vector for each word in a word association graph created with Small World of Words project (SWOWEN; Deyne et al., 2019) by propagating information related to masculine and feminine words $(w_m^i, w_f^i) \in \mathcal{L}$ using a random walk approach (Zhou et al., 2003). The gender information is represented as a 2-dimensional vector (b_m, b_f), where b_m and b_f denote respectively the masculine and feminine orientations of a word. The gender information vectors of masculine words, feminine words and other words are initialised respectively with vectors (1, 0), (0, 1) and (0, 0). The bias score of a word is defined as $\log(b_m/b_f)$. We evaluate the gender bias of word embeddings using the Pearson correlation coefficient between the bias score of each word and the score given by (11) computed as the averaged difference of cosine similarities between masculine and feminine words.

$$\frac{1}{|\mathcal{L}|}\sum_{i=1}^{|\mathcal{L}|}\left(f(w, w_m^i) - f(w, w_f^i)\right) \qquad (11)$$

SemBias: SemBias dataset (Zhao et al., 2018b) contains three types of word-pairs: (a) **Definition**, a gender-definition word pair (e.g. hero – heroine), (b) **Stereotype**, a gender-stereotype word pair (e.g., manager – secretary) and (c) **None**, two other word-pairs with similar meanings unrelated to gender (e.g., jazz – blues, pencil – pen). We use the cosine similarity between the $\overrightarrow{he} - \overrightarrow{she}$ gender directional vector and $\boldsymbol{a} - \boldsymbol{b}$ in above word pair (a, b) lists to measure gender bias. Zhao et al. (2018b) used a subset of 40 instances associated with 2 seed word-pairs, not used in the training split, to evaluate the generalisability of a debiasing method. For unbiased word embeddings, we expect high similarity scores in **Definition** category and low similarity scores in **Stereotype** and **None** categories.

WinoBias/OntoNotes: We use the WinoBias dataset (Zhao et al., 2018a) and OntoNotes (Weischedel et al., 2013) for coreference resolution to evaluate the effectiveness of our proposed debiasing method in a downstream task. WinoBias contains two types of sentences that require linking gendered pronouns to either male or female stereotypical occupations. In **Type 1**, co-reference decisions must be made using world knowledge about some given circumstances. However, in **Type 2**, these tests can be resolved using syntactic information and understanding of the pronoun. It involves two conditions: the pro-stereotyped (**pro**) condition links pronouns to occupations dominated by the gender of the pronoun, and the anti-stereotyped (**anti**) condition links pronouns to occupations not dominated by the gender of the pronoun. For a correctly debiased set of word embeddings, the difference between **pro** and **anti** is expected to be small. We use the model proposed by Lee et al. (2017) and implemented in AllenNLP (Gardner et al., 2017) as the coreference resolution method.

We used a bias comparing code[6] to evaluate **WEAT** dataset. Since the **WAT** code was not published, we contacted the authors to obtain the code and used it for evaluation. We used the evaluation code from GP-GloVe[7] to evaluate **SemBias** dataset. We used AllenNLP[8] to evaluate **WinoBias** and **OntoNotes** datasets. We used *evaluate_word_pairs* function and *evaluate_word_analogies* in gensim[9] to evaluate **word embedding benchmarks**.

5 Results

5.1 Overall Results

We initialise the word embeddings of the model by original (**Org**) and debiased (**Deb**) word embeddings and compare the coreference resolution accuracy using F1 as the evaluation measure.

In Table 1, we show the WEAT bias effects for cosine similarity and correlation on WAT dataset using the Pearson correlation coefficient. We see that the proposed method can significantly debias for various biases in all word embeddings in both WEAT and WAT. Especially in Word2Vec and fastText, almost all biases are debiased.

Table 2 shows the percentages where a word-pair is correctly classified as Definition, Stereotype or None. We see that our proposed method succesfully debiases word embeddings based on results on **Definition** and **Stereotype** in SemBias. In addition, we see that the SemBias-subset can be debiased for Word2Vec and fastText.

Table 3 shows the performance on WinoBias for **Type 1** and **Type 2** in **pro** and **anti** stereotypical

[6] https://github.com/hljames/compare-embedding-bias

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

[8] https://github.com/allenai/allennlp

[9] https://github.com/RaRe-Technologies/gensim

Embeddings	Word2Vec Org/Deb	GloVe Org/Deb	fastText Org/Deb
T1: flowers vs. insects	$1.46^{\dagger}$/$\mathbf{1.35}^{\dagger}$	$\mathbf{1.48}^{\dagger}$/$1.54^{\dagger}$	$1.29^{\dagger}$/$\mathbf{1.09}^{\dagger}$
T2: instruments vs. weapons	$1.56^{\dagger}$/$\mathbf{1.43}^{\dagger}$	$1.49^{\dagger}$/$\mathbf{1.41}^{\dagger}$	$1.56^{\dagger}$/$\mathbf{1.34}^{\dagger}$
T3: European vs. African American names	$0.46^{\dagger}$/$\mathbf{0.16}^{\dagger}$	$1.33^{\dagger}$/$\mathbf{1.04}^{\dagger}$	$0.79^{\dagger}$/$\mathbf{0.46}^{\dagger}$
T4: male vs. female	$1.91^{\dagger}$/$\mathbf{1.87}^{\dagger}$	$1.86^{\dagger}$/$\mathbf{1.85}^{\dagger}$	$1.65^{\dagger}$/$\mathbf{1.42}^{\dagger}$
T5: math vs. art	$0.85^{\dagger}$/$\mathbf{0.53}^{\dagger}$	$\mathbf{0.43}^{\dagger}$/$0.82^{\dagger}$	$1.14^{\dagger}$/$\mathbf{0.86}^{\dagger}$
T6: science vs. art	$1.18^{\dagger}$/$\mathbf{0.96}^{\dagger}$	$\mathbf{1.21}^{\dagger}$/$1.44^{\dagger}$	$1.16^{\dagger}$/$\mathbf{0.88}^{\dagger}$
T7: physical vs. mental conditions	0.90/**0.57**	1.03/**0.98**	0.83/**0.63**
T8: older vs. younger names	$-\mathbf{0.08}$/-0.10	$1.07^{\dagger}$/$\mathbf{0.92}^{\dagger}$	-0.32/$-\mathbf{0.13}$
T9: WAT	$0.48^{\dagger}$/$\mathbf{0.45}^{\dagger}$	$0.59^{\dagger}$/$\mathbf{0.58}^{\dagger}$	$0.54^{\dagger}$/$\mathbf{0.51}^{\dagger}$

Table 1: Rows T1-T8 show WEAT bias effects for the cosine similarity and row T9 shows the Pearson correlations on the WAT dataset with cosine similarity. † indicates bias effects that are insignificant at $\alpha < 0.01$.

Embeddings	Word2Vec Org/Deb	GloVe Org/Deb	fastText Org/Deb
definition	83.0/**83.9**	83.0/**83.4**	92.0/**93.2**
stereotype	13.4/**12.3**	12.0/**11.4**	5.5/**4.3**
none	**3.6**/3.9	**5.0**/5.2	**2.5**/**2.5**
sub-definition	50.0/**57.5**	**67.5**/**67.5**	82.5/**85.0**
sub-stereotype	40.0/**32.5**	**27.5**/**27.5**	12.5/**10.0**
sub-none	**10.0**/**10.0**	**5.0**/**5.0**	**5.0**/**5.0**

Table 2: Prediction accuracies for gender relational analogies on SemBias.

Embeddings	Word2Vec Org/Deb	GloVe Org/Deb	fastText Org/Deb
Type 1-pro	70.1/69.4	70.8/69.5	70.1/69.7
Type 1-anti	49.9/50.5	50.9/52.1	52.0/51.6
Avg	**60.0**/**60.0**	**60.9**/60.8	**61.1**/60.7
Diff	20.2/**18.9**	19.9/**17.4**	**18.1**/**18.1**
Type 2-pro	84.7/83.7	79.6/78.9	83.8/82.5
Type 2-anti	77.9/77.5	66.0/66.4	75.1/76.4
Avg	**81.3**/80.6	**72.8**/72.7	**79.5**/**79.5**
Diff	6.8/**6.2**	13.6/**12.5**	8.7/**6.1**
OntoNotes	62.6/**62.7**	62.5/**62.9**	63.3/**63.4**

Table 3: F1 on OntoNotes and WinoBias test set. WinoBias results have Type-1 and Type-2 in pro and anti stereotypical conditions. Average (Avg) and difference (Diff) of anti and pro stereotypical scores are shown.

conditions. In most settings, the diff is smaller for the debiased than the original word embeddings, which demonstrates the effectiveness of our proposed method. From the results for Avg, we see that debiasing is achieved with almost no loss in performance. In addition, the debiased scores on the OntoNotes are higher than the original scores for all word embeddings.

	GloVe	HD	GN-GloVe	GP-GloVe	Ours
T1	$0.89^{\dagger}$	$0.97^{\dagger}$	$1.10^{\dagger}$	$1.24^{\dagger}$	$\mathbf{0.74}^{\dagger}$
T2	$1.25^{\dagger}$	$1.23^{\dagger}$	$1.25^{\dagger}$	$1.31^{\dagger}$	$\mathbf{1.22}^{\dagger}$
T5	0.49	-0.40	**0.00**	0.21	0.35
T6	$1.22^{\dagger}$	**-0.11**	$1.13^{\dagger}$	$0.78^{\dagger}$	$1.05^{\dagger}$
T7	1.19	1.23	1.11	**1.01**	1.03

Table 4: WEAT bias effects for the cosine similarity on prior methods and proposed method. † indicates bias effects that are insignificant at $\alpha < 0.01$. T* are aligned with those in Table 1.

5.2 Comparison with Existing Methods

We compare the proposed method against the existing debiasing methods (Bolukbasi et al., 2016; Zhao et al., 2018b; Kaneko and Bollegala, 2019) mentioned in § 2 on WEAT, which contains different types of biases. We debias Glove[10], which is used in Zhao et al. (2018b). All word embeddings used in these experiments are the pre-trained word embeddings used in the existing debiasing methods. Words in evaluation sets T3, T4 and T8 are not covered by the input pre-trained embeddings and hence not considered in this evaluation. From Table 4 we see that only the proposed method debiases all biases accurately. T5 and T6 are the tests for gender bias; despite prior debiasing methods do well in those tasks, they are not able to address other types of biases. Notably, we see that the proposed method can debias more accurately compared to previous methods that use word lists for gender debiasing, such as Bolukbasi et al. (2016) in T5 and Zhao et al. (2018b) in T6.

5.3 Dominant Gloss vs All Glosses

In Table 5, we investigate the effect of using the dominant gloss (i.e. the gloss for the most frequent

[10] https://github.com/uclanlp/gn_glove

Embeddings	Word2Vec Dom/All	GloVe Dom/All	fastText Dom/All
definition	83.4/**83.9**	**83.9**/83.4	92.5/**93.2**
stereotype	12.7/**12.3**	11.8/**11.4**	4.8/**4.3**
none	**3.9**/**3.9**	**4.3**/5.2	2.7/**2.5**
sub-definition	55.0/**57.5**	**67.5**/**67.5**	77.5/**85.0**
sub-stereotype	35.0/**32.5**	**27.5**/**27.5**	12.5/**10.0**
sub-none	**10.0**/**10.0**	**5.0**/**5.0**	10.0/**5.0**

Table 5: Performance obtained when using only the dominant gloss (Dom) or all glosses (All) on SemBias.

Embeddings	Word2Vec Org/Deb	GloVe Org/Deb	fastText Org/Deb
WS	**62.4**/60.3	60.6/**68.9**	64.4/**67.0**
SIMLEX	44.7/**46.5**	39.5/**45.1**	44.2/**47.3**
RG	75.4/**77.9**	68.1/**74.1**	75.0/**79.6**
MTurk	63.1/**63.6**	62.7/**69.4**	67.2/**69.9**
RW	75.4/**77.9**	68.1/**74.1**	75.0/**79.6**
MEN	68.1/**69.4**	67.7/**76.7**	67.6/**71.8**
MSR	**73.6**/72.6	73.8/**75.1**	**83.9**/80.5
Google	**74.0**/73.7	76.8/**77.3**	**87.1**/85.7

Table 6: The Spearman correlation coefficients between human ratings and cosine similarity scores computed using word embeddings for the word pairs in semantic similarity benchmarks.

sense of the word) when creating $s(w)$ on SemBias benchmark as opposed to using all glosses (same as in Table 2). We see that debiasing using all glosses is more effective than using only the dominant gloss.

5.4 Word Embedding Benchmarks

It is important that a debiasing method removes only discriminatory biases and preserves semantic information in the original word embeddings. If the debiasing method removes more information than necessary from the original word embeddings, performance will drop when those debiased embeddings are used in NLP applications. Therefore, to evaluate the semantic information preserved after debiasing, we use semantic similarity and word analogy benchmarks as described next.

Semantic Similarity: The semantic similarity between two words is calculated as the cosine similarity between their word embeddings and compared against the human ratings using the Spearman correlation coefficient. The following datasets are used: Word Similarity 353 (**WS**; Finkelstein et al., 2001), **SimLex** (Hill et al., 2015), Rubenstein-Goodenough (**RG**; Rubenstein and Goodenough, 1965), **MTurk** (Halawi et al., 2012), rare words (**RW**; Luong et al., 2013) and **MEN** (Bruni et al., 2012).

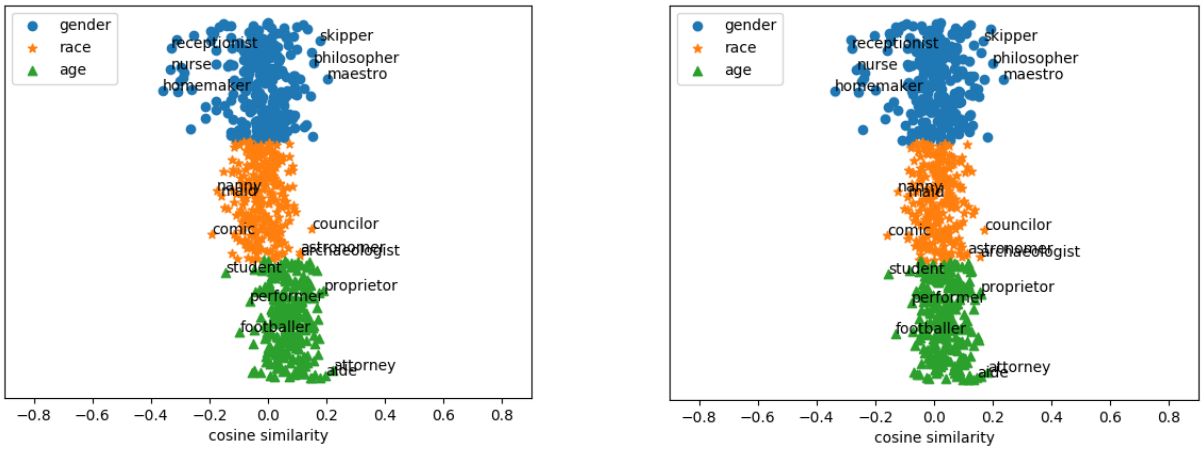
Word Analogy: In word analogy, we predict d that completes the proportional analogy "a is to b as c is to what?", for four words a, b, c and d. We use CosAdd (Levy and Goldberg, 2014), which determines d by maximising the cosine similarity between the two vectors ($\boldsymbol{b} - \boldsymbol{a} + \boldsymbol{c}$) and $\boldsymbol{d}$. Following Zhao et al. (2018b), we evaluate on **MSR** (Mikolov et al., 2013c) and **Google** analogy datasets (Mikolov et al., 2013a) as shown in Table 6.

From Table 6 we see that for all word embeddings, debiased using the proposed method accurately preserves the semantic information in the original embeddings. In fact, except for Word2Vec embeddings on WS dataset, we see that the accuracy of the embeddings have *improved* after the debiasing process, which is a desirable side-effect. We believe this is due to the fact that the information in the dictionary definitions is used during the debiasing process. Overall, our proposed method removes unfair biases, while retaining (and sometimes further improving) the semantic information contained in the original word embeddings.

We also see that for GloVe embeddings the performance has improved after debiasing whereas for Word2Vec and fastText embeddings the opposite is true. Similar drop in performance in word analogy tasks have been reported in prior work (Zhao et al., 2018b). Besides CosAdd there are multiple alternative methods proposed for solving analogies using pre-trained word embeddings such as CosMult, PairDiff and supervised operators (Bollegala et al., 2015, 2014; Hakami et al., 2018). Moreover, there have been concerns raised about the protocols used in prior work evaluating word embeddings on word analogy tasks and the correlation with downstream tasks (Schluter, 2018). Therefore, we defer further investigation in this behaviour to future work.

5.5 Visualising the Outcome of Debiasing

We analyse the effect of debiasing by calculating the cosine similarity between neutral occupational words and gender ($\overrightarrow{he} - \overrightarrow{she}$), race ($\overrightarrow{Caucasoid} - \overrightarrow{Negroid}$) and age ($\overrightarrow{elder} - \overrightarrow{youth}$) directions. The neutral occupational words list is based on Bolukbasi et al. (2016) and is listed in the Supplementary. Figure 1 shows the visualisation result for Word2Vec. We see that original Word2Vec shows some gender words are especially away from the



(a) Original Word2Vec

(b) Debiased Word2vec

Figure 1: Cosine similarity between neutral occupation words for vector directions on gender ($\overrightarrow{he} - \overrightarrow{she}$), race ($\overrightarrow{Caucasoid} - \overrightarrow{Negroid}$), and age ($\overrightarrow{elder} - \overrightarrow{youth}$) vectors.

origin (0.0). Moreover, age-related words have an overall bias towards "elder". Our debiased Word2Vec gathers vectors around the origin compared to the original Word2Vec for all gender, race and age vectors.

On the other hand, there are multiple words with high cosine similarity with the female gender after debiasing. We speculate that in rare cases their definition sentences contain biases. For example, in the WordNet the definitions for "homemaker" and "nurse" include gender-oriented words such as "a wife who manages a household while her husband earns the family income" and "a woman who is the custodian of children." It remains an interesting future challenge to remove biases from dictionaries when using for debiasing. Therefore, it is necessary to pay attention to biases included in the definition sentences when performing debiasing using dictionaries. Combining definitions from multiple dictionaries could potentially help to mitigate biases coming from a single dictionary. Another future research direction is to evaluate the proposed method for languages other than English using multilingual dictionaries.

6 Conclusion

We proposed a method to remove biases from pre-trained word embeddings using dictionaries, without requiring pre-defined word lists. The experimental results on a series of benchmark datasets show that the proposed method can remove unfair biases, while retaining useful semantic information encoded in pre-trained word embeddings.

References

Mohammed Alsuhaibani, Takanori Maehara, and Danushka Bollegala. 2019. Joint learning of hierarchical word embeddings from a corpus and a taxonomy. In *Proc. of the Automated Knowledge Base Construction Conference*.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A simple but tough-to-beat baseline for sentence embeddings. In *Proc. of ICLR*.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Danushka Bollegala, Takanori Maehara, and Ken-ichi Kawarabayashi. 2015. Embedding semantic relationas into word representations. In *Proc. of IJCAI*, pages 1222 – 1228.

Danushka Bollegala, Takanori Maehara, Yuichi Yoshida, and Ken-ichi Kawarabayashi. 2014. Learning word representations from relational graphs. In *Proc. of 29th AAAI Conference on Aritificial Intelligence (AAAI 2015)*, pages 2146 – 2152.

Danushka Bollegala, Alsuhaibani Mohammed, Takanori Maehara, and Ken-ichi Kawarabayashi. 2016. Joint word representation learning using a corpus and a semantic lexicon. In *Proc. of AAAI*, pages 2690–2696.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *NIPS*.

Shikha Bordia and Samuel R. Bowman. 2019. Identifying and reducing gender bias in word-level language models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for*

Computational Linguistics: Student Research Workshop, pages 7–15, Minneapolis, Minnesota. Association for Computational Linguistics.

Tom Bosc and Pascal Vincent. 2018. Auto-encoding dictionary definitions into consistent word embeddings. In *EMNLP*.

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)*, pages 136–145. Association for Computational Linguistics.

A. Budanitsky and G. Hirst. 2006. Evaluating wordnet-based measures of semantic distance. *Computational Linguistics*, 32(1):13–47.

Alexander Budanitsky and Graeme Hirst. 2001. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures. In *NAACL 2001*.

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

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.

Simon De Deyne, Danielle J. Navarro, Amy Perfors, Marc Brysbaert, and Gert Storms. 2019. The "small world of words" english word association norms for over 12,000 cue words. *Behavior Research Methods*, 51:987–1006.

Yupei Du, Yuanbin Wu, and Man Lan. 2019. Exploring human gender stereotypes with word association test. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6133–6143, Hong Kong, China. Association for Computational Linguistics.

Yanai Elazar and Yoav Goldberg. 2018. Adversarial Removal of Demographic Attributes from Text Data. In *Proc. of EMNLP*.

Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. 2019. Understanding undesirable word embedding associations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1696–1705, Florence, Italy. Association for Computational Linguistics.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado. Association for Computational Linguistics.

Christiane Fellbaum, editor. 1998. *WordNet: an electronic lexical database*. MIT Press.

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*, WWW '01, pages 406–414, New York, NY, USA. ACM.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Goran Glavaš and Ivan Vulić. 2018. Explicit retrofitting of distributional word vectors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 34–45. Association for Computational Linguistics.

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 Workshop on Widening NLP*, pages 60–63, Florence, Italy. Association for Computational Linguistics.

Anthony G. Greenwald, Debbie E. McGhee, and Jordan L. K. Schwatz. 1998. Measuring individual differences in implicit cognition: The implicit association test. *Journal of Personality and Social Psychology*, 74(6):1464–1480.

Huda Hakami, Kohei Hayashi, and Danushka Bollegala. 2018. Why does PairDiff work? - a mathematical analysis of bilinear relational compositional operators for analogy detection. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2493–2504, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

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*, KDD '12, pages 1406–1414, New York, NY, USA. ACM.

Rowan Hall Maudslay, Hila Gonen, Ryan Cotterell, and Simone Teufel. 2019. It's all in the name: Mitigating gender bias with name-based counterfactual data substitution. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language*

Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5266–5274, Hong Kong, China. Association for Computational Linguistics.

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

Jay J. Jiang and David W. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *10th Intl. Conf. Research on Computational Linguistics (ROCLING)*, pages 19 – 33.

Masahiro Kaneko and Danushka Bollegala. 2019. Gender-preserving debiasing for pre-trained word embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1641–1650, Florence, Italy. Association for Computational Linguistics.

Masahiro Kaneko and Danushka Bollegala. 2020. Autoencoding improves pre-trained word embeddings. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1699–1713, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. In *Proc. of ICLR*.

Anja Lambrecht and Catherine E. Tucker. 2016. Algorithmic bias? an empirical study into apparent gender-based discrimination in the display of stem career ads. *SSRN Electronic Journal*.

Anne Lauscher and Goran Glavas. 2019. Are we consistently biased? multidimensional analysis of biases in distributional word vectors. In **SEM@NAACL-HLT*.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, Copenhagen, Denmark. Association for Computational Linguistics.

Omer Levy and Yoav Goldberg. 2014. Linguistic regularities in sparse and explicit word representations. In *CoNLL*.

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. Association for Computational Linguistics.

Thang Luong, Richard Socher, and Christopher Manning. 2013. Better word representations with recursive neural networks for morphology. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 104–113. Association for Computational Linguistics.

Thomas Manzini, Lim Yao Chong, Alan W Black, and Yulia Tsvetkov. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. 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 615–621, Minneapolis, Minnesota. Association for Computational Linguistics.

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.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013b. Linguistic regularities in continous space word representations. In *NAACL-HLT*, pages 746 – 751.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013c. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Jeffery Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: global vectors for word representation. In *EMNLP*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proc. of NAACL-HLT*.

Shauli Ravfogel, Yanai Elazar, Hila Gonen, Michael Twiton, and Yoav Goldberg. 2020. Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection. In *Proc. of ACL*.

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

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14. Association for Computational Linguistics.

Natalie Schluter. 2018. The word analogy testing caveat. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 242–246. Association for Computational Linguistics.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2019. The woman worked as a babysitter: On biases in language generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3405–3410, Hong Kong, China. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15:1929–1958.

Julien Tissier, Christopher Gravier, and Amaury Habrard. 2017. Dict2vec : Learning word embeddings using lexical dictionaries. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Copenhagen, Denmark. Association for Computational Linguistics.

Eva Vanmassenhove, Christian Hardmeier, and Andy Way. 2018. Getting gender right in neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3003–3008. Association for Computational Linguistics.

Jesse Vig. 2019. Visualizing Attention in Transformer-Based Language Representation Models.

Koki Washio, Satoshi Sekine, and Tsuneaki Kato. 2019. Bridging the defined and the defining: Exploiting implicit lexical semantic relations in definition modeling. In *EMNLP/IJCNLP*.

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.

Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. 2017. Controllable invariance through adversarial feature learning. In *Proc. of NIPS*.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. 2019. Gender bias in contextualized word embeddings. 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 629–634, Minneapolis, Minnesota. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2941–2951, Copenhagen, Denmark. Association for Computational Linguistics.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018a. Gender bias in coreference resolution: Evaluation and debiasing methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20. Association for Computational Linguistics.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018b. Learning Gender-Neutral Word Embeddings. In *Proc. of EMNLP*, pages 4847–4853.

Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. 2003. Learning with local and global consistency. In *NIPS*.

Ran Zmigrod, Sebastian J. Mielke, Hanna Wallach, and Ryan Cotterell. 2019. Counterfactual Data Augmentation for Mitigating Gender Stereotypes in Languages with Rich Morphology. In *Proc. of ACL*.