Abstract

Online texts—across genres, registers, domains, and styles—are riddled with human stereotypes, expressed in overt or subtle ways. Word embeddings, trained on these texts, perpetuate and amplify these stereotypes, and propagate biases to machine learning models that use word embeddings as features. In this work, we propose a method to debias word embeddings in multiclass settings such as race and religion, extending the work of (Bolukbasi et al., 2016) from the binary setting, such as binary gender. Next, we propose a novel methodology for the evaluation of multiclass debiasing. We demonstrate that our multiclass debiasing is robust and maintains the efficacy in standard NLP tasks.

1 Introduction

In addition to possessing informative features useful for a variety of NLP tasks, word embeddings reflect and propagate social biases present in training corpora (Caliskan et al., 2017; Garg et al., 2018). Machine learning systems that use embeddings can further amplify biases (Barocas and Selbst, 2016; Zhao et al., 2017), discriminating against users, particularly those from disadvantaged social groups.

(Bolukbasi et al., 2016) introduced a method to debias embeddings by removing components that lie in stereotype-related embedding subspaces. They demonstrate the effectiveness of the approach by removing gender bias from word2vec embeddings (Mikolov et al., 2013), preserving the utility of embeddings and potentially alleviating biases in downstream tasks. However, this method was only for binary labels (e.g., male/female), whereas most real-world demographic attributes, including gender, race, religion, are not binary but continuous or categorical, with more than two categories.

In this work, we show a generalization of Bolukbasi et al.’s (2016) which enables multiclass debiasing, while preserving utility of embeddings (§3). We train word2vec embeddings using the Reddit L2 corpus (Rabinovich et al., 2018) and apply multiclass debiasing using lexicons from studies on bias in NLP and social science (§4.2). We introduce a novel metric for evaluation of bias in collections of word embeddings (§5). Finally, we validate that the utility of debiased embeddings in the tasks of part-of-speech (POS) tagging, named entity recognition (NER), and POS chunking is on par with off-the-shelf embeddings.

2 Background

As defined by (Bolukbasi et al., 2016), debiasing word embeddings in a binary setting requires identifying the bias subspace of the embeddings. Components lying in that subspace are then removed from each embedding.

2.1 Identifying the bias subspace

(Bolukbasi et al., 2016) define the gender subspace using defining sets of words, where the words in
each set represent different ends of the bias. For example, in the case of gender, one defining set might be the gendered pronouns \{he, she\} and another set might be the gendered nouns \{man, woman\}. The gender subspace is then computed from these defining sets by 1) computing the vector differences of the word embeddings of words in each set from the set’s mean, and 2) taking the most significant components of these vectors.

### 2.2 Removing bias components

Following the identification of the gender subspace, one can apply hard or soft debiasing (Bolukbasi et al., 2016) to completely or partially remove the subspace components from the embeddings.

#### Hard debiasing

Hard debiasing (also called “Neutralize and Equalize”) involves two steps. First, bias components are removed from words that are not gendered and should not contain gender bias (e.g., doctor, nurse), and second, gendered word embeddings are centered and their bias components are equalized. For example, in the binary case, man and woman should have bias components in opposite directions, but of the same magnitude. Intuitively, this then ensures that any neutral words are equidistant to any biased words with respect to the bias subspace.

More formally, to neutralize, given a bias subspace \(B\) spanned by the vectors \(\{b_1, b_2, ..., b_k\}\), we compute the component of each embedding in this subspace:

\[
 w_B = \sum_{i=1}^{k} \langle w, b_i \rangle b_i \tag{1}
\]

We then remove this component from words that should be bias-neutral and normalize to get the debiased embedding:

\[
 w' = \frac{w - w_B}{\|w - w_B\|} \tag{2}
\]

To equalize the embeddings of words in an equality set \(E\), let \(\mu = \frac{1}{|E|} \sum_{w \in E} w\) be the mean embedding of the words in the set and \(\mu_B\) be its component in the bias subspace as calculated in Equation 1. Then, for \(w \in E\),

\[
 w' = (\mu - \mu_B) + \sqrt{1 - \|\mu - \mu_B\|^2} \frac{w_B - \mu_B}{\|w_B - \mu_B\|} \tag{3}
\]

Note that in both Equations 2 and 3, the new embedding has unit length.

#### Soft debiasing

Soft debiasing involves learning a projection of the embedding matrix that preserves the inner product between biased and debiased embeddings while minimizing the projection onto the bias subspace of embeddings that should be neutral.

Given embeddings \(W\) and \(N\) which are embeddings for the whole vocabulary and the subset of bias-neutral words respectively, and the bias subspace \(B\) obtained in Section 2.1, soft debiasing seeks for a linear transformation \(A\) that minimizes the following objective:

\[
 \| (AW)^\top (AW) - W^\top W \|_F^2 + \lambda \| (AN)^\top (AB) \|_F^2 \tag{4}
\]

Minimizing the first term preserves the inner product after the linear transformation \(A\), and minimizing the second term minimizes the projection onto the bias subspace \(B\) of embeddings. \(\lambda \in \mathbb{R}\) is a tunable parameter that balances the two objectives.

### 3 Methodology

We now discuss our proposed extension of word embedding debiasing to the multiclass setting.

#### 3.1 Debiasing

As in the binary setting, debiasing consists of two steps: identifying the “bias subspace” and removing this component from the set of embeddings.

#### Identifying the bias subspace

The core contribution of our work is in identifying the “bias subspace” in a multiclass setting; if we can identify the bias subspace then prior work can be used for multiclass debiasing.

Past work has shown that it is possible to linearly separate multiple social classes based on components of word embeddings (Garg et al., 2018). Based on this we hypothesize that there exists some component of these embeddings which can capture multiclass bias. While a multiclass problem is inherently not a linearly separable problem, a one versus rest classifier is. Following from this, the computation of a multiclass bias subspace does not have any linear constraints, though it does come with a loss of resolution. As a result we can compute the principal components
Table 2: The associated mean average cosine similarity (MAC) (defined in Section 3.2) and p-Values for debiasing methods for gender, race, and religious bias.

<table>
<thead>
<tr>
<th>Biasing Component</th>
<th>MAC</th>
<th>p-Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Biased</td>
<td>0.623</td>
<td>N/A</td>
</tr>
<tr>
<td>Hard Debiased</td>
<td>0.700</td>
<td>3.2e-10</td>
</tr>
<tr>
<td>Soft Debiased (λ = 0.2)</td>
<td>0.747</td>
<td>1.711e-12</td>
</tr>
<tr>
<td>Gender Debiasing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Race Debiasing</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Religion Debiasing</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

required to compute the “bias subspace” by simply adding an additional term for each additional bias class to each defining set.

Formally, given defining sets of word embeddings \( D_1, D_2, ..., D_n \), let the mean of the defining set \( i \) be \( \mu_i = \frac{1}{|D_i|} \sum_{w \in D_i} w \), where \( w \) is the word embedding of \( w \). Then the bias subspace \( B \) is given by the first \( k \) components of the following principal component analysis (PCA) evaluation:

\[
\text{PCA} \left( \bigcup_{i=1}^n \bigcup_{w \in D_i} w - \mu_i \right) \tag{5}
\]

The number of components \( k \) can be empirically determined by inspecting the eigenvalues of the PCA, or using a threshold. Also, note that the defining sets do not have to be the same size. We discuss the robustness of this method later.

Removing Bias Components

Following the identification of the bias subspace, we apply the hard Neutralize and Equalize debiasing and soft debiasing method presented in (Bolukbasi et al., 2016) and discussed in Section 2.2 to completely or partially remove the subspace components from the embeddings.

For equalization, we take the defining sets to be the equality sets as well.

3.2 Quantifying Bias Removal

We propose a new metric for the evaluation of bias in collections of words which is simply the mean average cosine similarity (MAC). This approach is motivated by the WEAT evaluation method proposed by (Caliskan et al., 2017) but modified for a multiclass setting. To compute this metric the following data is required: a set of target word embeddings \( T \) containing terms that inherently contain some form of social bias (e.g. \{church, synagogue, mosque\}), and a set \( A \) which contains sets of attributes \( A_1, A_2, ..., A_N \) containing word embeddings that should not be associated with any word embeddings contained in the set \( T \) (e.g. \{violent, liberal, conservative\}).

We define a function \( S \) that computes the mean cosine distance between a particular target \( T_i \) and all terms in a particular attribute set \( A_j \):

\[
S(t, A_j) = \frac{1}{N} \sum_{a \in A_j} \cos(t, a), \tag{6}
\]

where the cosine distance is:

\[
\cos(u, v) = 1 - \frac{u \cdot v}{\|u\|_2 \cdot \|v\|_2}. \tag{7}
\]

Finally, we define MAC as:

\[
\text{MAC}(T, A) = \frac{1}{|T||A|} \sum_{T_i \in T} \sum_{A_j \in A} S(T_i, A_j) \tag{8}
\]

We also perform a paired \( t \)-test on the distribution of average cosines used to calculate the MAC. Thus we can quantify the effect of debiasing on word embeddings in \( T \) and sets in \( A \).

3.3 Measuring Downstream Utility

To measure the utility of the debiased word embeddings, we use the tasks of NER, POS tagging, and POS chunking. This is to ensure that the debiasing process has not destroyed the utility of the word embeddings. We evaluate test sentences that contain at least one word affected by debiasing. Additionally, we measure the change in performance after replacing the biased embedding matrix by a debiased one, and retraining the model on debiased embeddings.

4 Data

In this section we discuss the different data sources we used for our initial word embeddings, the social bias data used for evaluating bias, and the linguistic data used for evaluating the debiasing process.

4.1 Embedding Language Corpus

We used the L2-Reddit corpus (Rabinovich et al., 2018), a collection of Reddit posts and comments
by both native and non-native English speakers. The native countries of post authors are determined based on their posts in country- and region-specific subreddits (such as r/Europe and r/UnitedKingdom), and other metadata such as user flairs, which serve as self-identification of the user’s country of origin.

In this work, we exclusively explore data collected from the United States. This was done to leverage extensive studies of social bias done in the United States. To obtain the initial biased word embeddings, we trained word2vec embeddings on several downstream tasks. Specifically, the CoNLL 2003 shared task (Tjong Kim Sang and De Meulder, 2003) which provides evaluation data for NER, POS tagging, and POS chunking.

### Table 3: The performance of embeddings the downstream tasks of NER, POS Tagging, and POS Chunking.

<table>
<thead>
<tr>
<th>Model Retraining</th>
<th>Hard Gender Debiasing</th>
<th>Hard Racial Debiasing</th>
<th>Hard Religious Debiasing</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NER Tagging</td>
<td>POS Tagging</td>
<td>POS Chunking</td>
</tr>
<tr>
<td>Biased F1</td>
<td>0.9614</td>
<td>0.9616</td>
<td>0.9952</td>
</tr>
<tr>
<td>Δ F1</td>
<td>+0.0047</td>
<td>+0.0045</td>
<td>+0.0053</td>
</tr>
<tr>
<td>Δ Precision</td>
<td>-0.0202</td>
<td>-0.0198</td>
<td>+0.0156</td>
</tr>
<tr>
<td>Δ Recall</td>
<td>+0.0165</td>
<td>+0.0208</td>
<td>+0.0156</td>
</tr>
</tbody>
</table>

For gender, we used vocabularies created by (Bolukbasi et al., 2016) and (Caliskan et al., 2017).

For race we consulted a number of different sources for each race: Caucasians (Chung-Herrera and Lankau, 2005; Goad, 1998); African Americans (Punyanunt-Carter, 2008; Brown Givens and Monahan, 2005; Chung-Herrera and Lankau, 2005; Hakonen, 1995; Welch, 2007; Kawai, 2005); and Asian Americans (Leong and Hayes, 1990; Lin et al., 2005; Chung-Herrera and Lankau, 2005; Osajima, 2005; Garg et al., 2018).

Finally, for religion we used the following sources and labels: Christians (Rios et al., 2015; Zuckermand, 2009; Unnever et al., 2005); Jews (Dundes, 1971; Fetzer, 2000); and Muslims (Shryock, 2010; Alsultany, 2012; Shaheen, 1997).

### 4.3 Downstream Tasks

We evaluate biased and debiased word embeddings on several downstream tasks. Specifically, the CoNLL 2003 shared task (Tjong Kim Sang and De Meulder, 2003) which provides evaluation data for NER, POS tagging, and POS chunking.

### 5 Results and Discussion

In this section we review the results of our experiments and discuss what those results mean in the context of this work.

### 5.1 Observations of Bias

We use the analogy task from (Bolukbasi et al., 2016) to demonstrate that bias exists in these word embeddings. In order to construct our analogies we trained five word2vec (Mikolov et al., 2013) embedding spaces on the same data. We then constructed a set of analogies for each embedding space taking the intersection of these to form a working set of analogies. We performed this extra step in order to ensure the analogies were robust to perturbations in the embedding space. Following this analysis we observe that bias is present in generated analogies by viewing them directly.
small subset of these analogies are in Table 1 to highlight our findings.

5.2 Removal of Bias
We perform our debiasing in the same manner as described in Section 3.1 and calculate the MAC scores and $p$-values to measure the effects of debiasing. Results are presented in Table 2.

Does multiclass debiasing decrease bias? We see that this debiasing procedure categorically moves MAC scores closer to 1.0. This indicates an increase in cosine distance. Further, the associated $p$-values indicate these changes are statistically significant. This demonstrates that our approach for multiclass debiasing decreases bias.

5.3 Downstream Effects of Bias Removal
The effects of debiasing on downstream tasks are shown in Table 3. Debiasing can either help or harm performance. For POS tagging there is almost always a decrease in performance. However, for NER and POS chunking, there is a consistent increase. We conclude that these models have learned to depend on some bias subspaces differently. Note that many performance changes are of questionable statistical significance.

Does multiclass debiasing preserve semantic utility? We argue the minor changes in Table 3 support the preservation of semantic utility in the multiclass setting, especially compared to gender debiasing which is known to preserve utility (Bolukbasi et al., 2016).

Is the calculated bias subspace robust? The bias subspace is at least robust enough to support the above debiasing operations. This is shown by statistically significant changes in MAC scores.

6 Limitations & Future Work
Calculating multiclass bias subspace using our proposed approach has drawbacks. For example, in the binary gender case, the extremes of bias subspace reflect extreme male and female terms. However, this is not possible when projecting multiple classes into a linear space. Thus, while we can calculate the magnitude of the bias components, we cannot measure extremes of each class.

Additionally, the methods presented here rely on words that represent biases (defining sets) and words that should or should not contain biases (equality sets). These lists are based on data collected specifically from the US. Thus, they may not translate to other countries or cultures. Further, some of these vocabulary terms, while peer reviewed, may be subjective and may not fully capture the bias subspace.

Recent work by Gonen and Goldberg (2019) suggests that debiasing methods based on bias component removal are insufficient to completely remove bias in the embeddings, since embeddings with similar biases are still clustered together after bias component removal. Following Gonen and Goldberg’s (2019) procedure, we plot the number of neighbors of a particular bias class as a function of the original bias, before and after debiasing in Figure 1 and 2 in the Appendix. In line with Gonen and Goldberg’s (2019) findings, simply removing the bias component is insufficient to remove multiclass “cluster bias”. However, increasing the size of the bias subspace reduces the correlation of the two variables (Table 4 in the Appendix).

7 Conclusion
We showed that word embeddings trained on www.reddit.com data contain multiclass biases. We presented a novel metric for evaluating debiasing procedures for word embeddings. We robustly removed multiclass bias using a generalization of existing techniques. Finally, we showed that this multiclass generalization preserves the utility of embeddings for different NLP tasks.

Acknowledgments
This research was supported by Grant No. IIS1812327 from the United States National Science Foundation (NSF). We also acknowledge several people who contributed to this work: Benjamin Pall for his valuable early support of this work; Elise Romberger who helped edit this work prior to its final submission. Finally, we are greatly appreciative of the anonymous reviewers for their time and constructive comments.

References
Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to


A Addressing Cluster Bias

To visualize the degree of cluster bias before and after our debiasing procedure, we follow a similar procedure to Gonen and Goldberg (2019). For
Table 4: Pearson’s $r$ and Spearman’s $\rho$ correlation coefficients between the number of biased neighbors and the original bias of professions with respect to target classes for religion. $k$ is the dimension of the bias subspace used ($k = 0$ represents the original embedding). All correlation coefficients have $p$-values $< 10^{-30}$.

<table>
<thead>
<tr>
<th>Target</th>
<th>$k$</th>
<th>$r$</th>
<th>$\rho$</th>
</tr>
</thead>
<tbody>
<tr>
<td>jew</td>
<td>0</td>
<td>0.767</td>
<td>0.875</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0.795</td>
<td>0.891</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.718</td>
<td>0.756</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.736</td>
<td>0.772</td>
</tr>
<tr>
<td>christian</td>
<td>0</td>
<td>0.925</td>
<td>0.947</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0.835</td>
<td>0.841</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.825</td>
<td>0.831</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.832</td>
<td>0.839</td>
</tr>
<tr>
<td>muslim</td>
<td>0</td>
<td>0.858</td>
<td>0.894</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0.774</td>
<td>0.812</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>0.715</td>
<td>0.721</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>0.712</td>
<td>0.718</td>
</tr>
</tbody>
</table>

FIGURE 1: Plots of number of neighbors to jew for each profession as a function of its original bias with respect to jew, before and after debiasing, for different subspace dimensionalities $k$.

---

2https://github.com/tolga-b/debiaswe/blob/master/data/professions.json
Figure 2: Plots of number of neighbors to *Muslim* for each profession as a function of its original bias with respect to *Muslim*, before and after debiasing, for different subspace dimensionalities $k$. 