Analyzing Bias in Word Embeddings

Derek Leung **Stephanie Shi** **Marshall Vail** **Wei Zhang**
{dcleung, stephshi, vailm, wzha}@seas.upenn.edu

Abstract

We examine bias present in word embeddings, in which every word is represented as a vector, an approach commonly used in machine learning and natural language processing tasks. We use the approach in (Bolukbasi et al., 2016) to remove gender bias in Google News word2vec and GloVe word embeddings. We find that based on the evaluation metrics used in literature that the debiasing algorithm in (Bolukbasi et al., 2016) is able to remove a significant portion of the gender bias from these word embeddings. We also explore political bias by training our own word embeddings on data from biased news sources and use the same algorithm to remove political bias from these word embeddings.

1 Introduction

Word embedding models are used frequently to solve various NLP problems. However, since these word embeddings are trained on data that is inherently biased, these resulting models preserve undesirable bias, such as gender and racial bias. For example, though the word "programmer" is gender neutral, an embedding model can find the following on an analogy-solving task:

$$\overrightarrow{\text{man}} - \overrightarrow{\text{woman}} \approx \overrightarrow{\text{programmer}} - \overrightarrow{\text{homemaker}}$$

meaning that programmer is closer to the vector for man than for woman, though we would expect the vector for programmer to be equidistant from that for man and that for woman. This poses a significant problem when these biased models are used, as they would only serve to further perpetuate gender and racial bias and stereotypes.

Thus, our goals are the following:

1. How much gender bias is currently present in various word embeddings commonly used in NLP and computational linguistic tasks?
2. How can we remove bias from these word embeddings while preserving their quality on various NLP tasks?
3. What are other biases not typically explored in literature present in these word embeddings?

We picked this task for our term project because of the importance that word embeddings play in various NLP tasks, and bias was also a topic that our team was collectively interested in. We also felt that we could produce a well-scoped out project for the given timeframe, as well as the high availability of data and word embeddings.

2 Literature Review

Much work has been done in this area. Below is a short summary of some of the seminal works.

2.1 Man is to Computer Programmer as Women is to Homemaker? Debiasing Word Embeddings

(Bolukbasi et al., 2016) is one of the first seminal works in the area of debiasing word vectors. The researchers define direct bias as the sum of the absolute cosine similarity between the gender neutral words and a learned gender direction g; g is roughly learned by combining several distances such as the differences between the vectors of she and he, woman and man, etc. Indirect bias is measured roughly by measuring the contribution of gender to each word vector; the gender component is defined as the similarity between two word vectors. To debias the pairs that should be gender neutral, the researchers subtracted the projection

onto the gender direction of these gender neutral word pairs.

We choose to implement the work done in this paper since it serves as a good baseline to compare both other work in this area and our extensions to. Furthermore, the methodology used in this paper is relatively straightforward, so implementing the work done here will serve as a good stepping stone to extending their research.

2.2 Semantic Derived Automatically From Language Corpora Contain Human-Like Biases

In (Caliskan et al., 2017), researchers define another metric called Word-Embedding Association Test (WEAT) to measure bias in word embeddings (specifically GloVe). The researchers use cosine similarity between a pair of vectors as analogous to reaction time in the Implicit Association Test (IAT). The researchers find that there were implicit gender-occupation biases, and suggest that there are historic biases contained within text corpora. The researchers highlight the importance of understanding bias and prejudice in making algorithms and other technologies that impact people.

2.3 Word Embeddings Quantify 100 Years of Gender and Ethnic Stereotypes

In (Garg et al., 2018), researchers analyze different word embeddings, including word2vec and COHA, trained over 100 years of text corpora to understand how gender and ethnic stereotypes have changed over time. To do so, they measure the strength of the association between neutral words and a group. Their metric for measuring bias is the relative norm difference, which is computed by first finding the representative word vector by taking the average of the vectors for each word in the given gender / ethnic group, and then the average Euclidean distance between each representative group vector and each vector of the neutral words. Ordinary least-squares regression is used to measure associations. The researchers found that bias generally moved in the direction they expected; for instance, the researchers found that gender bias in occupations moved toward 0 from the 1950s to the 1990s, which is in line with the gender movement.

2.4 Lipstick on a Pig

Hila Gonen and Yoav Goldberg (Gonen and Goldberg, 2019) show that the work done in (Bolukbasi et al., 2016) only superficially removes the debiasing, as the "bias ... is ingrained much more deeply in the embeddings space." Gonen and Goldberg do so by clustering the most biased words in the original vocabulary using k-means, and find that the clusters still align with gender with a high accuracy. Thus, Gonen and Goldberg posit that existing metrics that have been explored in research may be good at measuring bias, but are not necessarily effective as a baseline for comparison for debiasing. We include this paper since it highlights some of the shortcomings in current debiasing research.

3 Experimental Design

3.1 Data

We use various context-free word embeddings, including Google News word2vec, which is trained on roughly 100 billion words from a Google News dataset and has a vocabulary of 3 million words and phrases, as well as GloVe embeddings, which is trained on roughly 2 billion tweets from Twitter. For our last extension, we also attempted to use pre-trained BERT word embeddings. For debiasing, we use dictionary-retrieved gender-definitional words & occupations and crowdsourced gender-specific and equivalent words. To evaluate the performance of debiased word embeddings, we use various NLP task test data, namely the WordSimilarity-353 collection and Google analogy test set.

Our lists of gender words comes from those used in (Bolukbasi et al., 2016) and their Github repository[1]. We debias with respect to these lists of gender definitions and gender specific words. We debias word embeddings trained by Google on a large corpus of news articles `GoogleNews-vectors-negative300`.

3.2 Evaluation Metrics

Our evaluation metrics for our embeddings fall under two categories: measuring bias and measuring quality.

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

3.2.1 Measuring Bias

In order to measure bias, we employ a variety of different metrics used in other existing work. In particular, we decided to use the direct bias and indirect bias from (Bolukbasi et al., 2016), and Pearson correlation and clustering accuracy as in (Gonen and Goldberg, 2019).

Direct bias is a measurement of bias over a set N of gender neutral words. We find that given a learned vector g representing the gender subspace, we can measure the bias of any given word with embedding $\vec{w}$ using its similarity to g.

$$\text{DirectBias}_c = \frac{1}{|N|} \sum_{w \in N} |\cos(\vec{w}, g)|^c$$

Note that the variable c, as described in (Bolukbasi et al., 2016), is a lenience parameter (i.e. when $c = 0$, the measurement is very strict).

Indirect bias, on the other hand, is a measurement of bias given a pair of words with embeddings w and v, $\beta(w, v)$. This metric essentially describes what fraction of the similarity between the two words is a result of a shared alignment with a specific gender. In particular, we first decompose each embedding into its projection onto the gender vector g, and its component perpendicular to g.

$$w_g = (w \cdot g)g \qquad w_\perp = w - w_g$$

We then compare the similarity of the original embeddings with their similarity when their projection on the gender direction $\vec{g}$ is removed.

$$\beta(w, v) = \frac{\left(w \cdot v - \frac{w_\perp \cdot v_\perp}{\|w_\perp\|_2 \|v_\perp\|_2}\right)}{w \cdot v}$$

Since we also seek to understand how our embeddings change after applying any debiasing technique, we measure the correlation between our original embeddings and debiased embeddings using Pearson Correlation.

As done in (Gonen and Goldberg, 2019), we also consider the 1000 most biased words (500 in either direction for female and male) according to our embeddings and our learned gender direction g, and observe how easily these words can be clustered. To do this, we simply cluster the words using K-means with 2 clusters, and measure the accuracy of the resulting labels.

3.2.2 Measuring Quality

After applying any debiasing algorithm, we naturally change the geometry of our embeddings. Thus, it is important to consider how the quality of our embeddings change after such alterations. In order to measure the quality of the embeddings we consider its performance on simple NLP tasks. In particular, we consider the word similarity task and analogy-solving task.

Under the word similarity task, we take our list of word pairs with human-assigned similarity ratings, and then evaluate their similarities using our embeddings with cosine similarity. We can then measure the correlation between these similarity ratings using the Spearman Rank Correlation, which is a nonparametric measure of rank correlation.

For the analogy-solving task, we take our list of word tuples that form analogies, and have our embeddings attempt to solve these analogies by taking the word with the largest cosine similarity. The embeddings performance is measured by its ability to accurately predict the solution to the analogy.

3.3 Simple Baseline

Without any debiasing, we observe the following results in figure and tables below, which are similar to those reported in (Bolukbasi et al., 2016) and (Gonen and Goldberg, 2019):

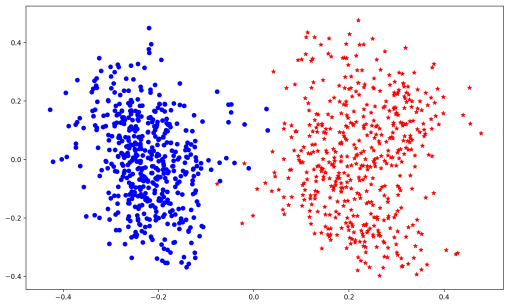

Figure 1: Clustering 1000 Most Biased Words (Original Google News word2vec)

We can see that there is a high degree of gender bias present in the Google News word2vec embeddings. This is demonstrated in all facets of our bias evaluations. Specifically, we find that the indirect bias and clustering accuracy to be particu-

Male	Female
journeyman	petite
burly	bra
rookie	sassy
hero	pageant
veteran	pregnant

Table 1: Top 5 Most Biased Words (Original Google News word2vec)

Direct Bias	0.08
Clustering Accuracy	0.994
Spearman Correlation for WS	0.6857
Google Analogies Accuracy	0.65

Table 2: Bias and Quality Evaluation (Original Google News word2vec)

Word Pairs	Indirect Bias
receptionist, softball	0.6723
waitress, softball	0.3178
homemaker, softball	0.3837

Table 3: Indirect Bias (Original Google News word2vec)

larly high. Additionally, by extracting the most biased words in these embeddings, we are able to observe blatant forms of gender bias in words like "petite" and "sassy", and "hero" and "veteran". The implication being that being petite or sassy is a stereotypically female-trait, while being a hero or a veteran is stereotypically male. Finally, in regard to the quality of the embeddings, we find that they perform quite well in both the word similarity and analogy-solving task, which is to be expected given the size and content of the training data for these embeddings.

4 Experimental Results

4.1 Published Baseline

4.1.1 Debiasing Algorithm

In order to debias our embeddings, we employ the technique used in (Bolukbasi et al., 2016). This algorithm can be roughly split into three main sections: identifying the gender subspace, neutralizing embeddings, and equalizing embeddings.

In order to identify the gender subspace we take our definitional pairs data set, and for each definitional pair, we find the gender difference and use PCA to find the main gender component.

In the neutralization step, we ensure that gender neutral words are zero in the gender subspace. In particular, we accomplish this by subtracting the projection of each vector onto gender direction (then normalizing). So, for any gender neutral word vector w, under the neutralization step it gets reembedded to

$$w' = \frac{w - w_g}{||w - w_g||}$$

In the final step of our debiasing algorithm, the equalize step, we equalize pairs of words outside of the subspace, which enforces that any gender neutral word is equidistant to both words in each equality pair. So, for any equality pair of embeddings (w, v), we first take the average of the two embeddings μ, and under the equalization step each gets reembedded to

$$w' = \mu_\perp + \sqrt{1 - ||\mu_\perp||^2} \frac{w_g - \mu_g}{||w_g - \mu_g||}$$

To clarify by example what this step's purpose is, consider the equality set of ("grandmother", "grandfather"), and "babysit" as a gender-neutral word for which we wish to equalize with respect to. We first want "grandmother" and "grandfather" to be equidistant to "babysit", but second we notice that presumably both words are closer to "babysit" than another equality pair of ("spokeswoman", "spokesman"). Thus, we also seek to ensure that we preserve that the first equality set is closer to "babysit" than the second.

4.1.2 Baseline Results

In this subsection we discuss the results for both bias and quality measurement we observed for the word2vec Google News embeddings after applying our debiasing algorithm. The results are shown in figure and tables below.

Direct Bias	0.015
Clustering Accuracy	0.858
Spearman Correlation for WS	0.6826
Google Analogies Accuracy	0.71

Table 4: Bias and Quality Evaluation (Debiased Google News word2vec)

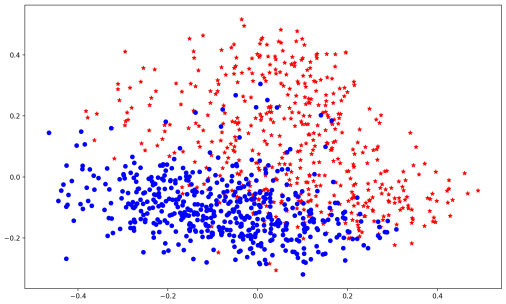

Figure 2: Clustering 1000 Most Biased Words[2] (Debiased Google News word2vec)

Word Pairs	Indirect Bias
receptionist, softball	5.27e-15
waitress, softball	-0.047
homemaker, softball	6.97e-15

Table 5: Indirect Bias (Debiased Google News word2vec)

It is clear from our results that our algorithm successfully debiases the embeddings with respect to direct bias and indirect bias (both have been essentially reduced to zero). As pointed out in (Gonen and Goldberg, 2019), however, we are still able to recover biases in the embeddings through clustering. In particular, we find that the clustering accuracy of the most biased words in the original embeddings still sits at a relatively high 86 percent. It is important to note, though, that our debiasing algorithm did not diminish the quality of the embeddings in terms of the word similarity and analogy-solving tasks.

4.2 Extension 1: Gender Bias in GloVe

For our first extension, we wanted to explore gender bias and the effectiveness of our debiasing techniques on other popular pre-trained word embeddings. Specifically, we decided to use GloVe embeddings trained on Twitter data. By experimenting with embeddings trained on social media data, we hoped to gain a stronger understanding of the prevalence of gender bias in society in the age of the internet.

Using similar methods as our exploration into the Google News word2vec embeddings, we found the the observe the following results for the original GloVe embeddings for Twitter in the figure and table below. These are similar to the ones reported in (Bolukbasi et al., 2016) and (Gonen and Goldberg, 2019).

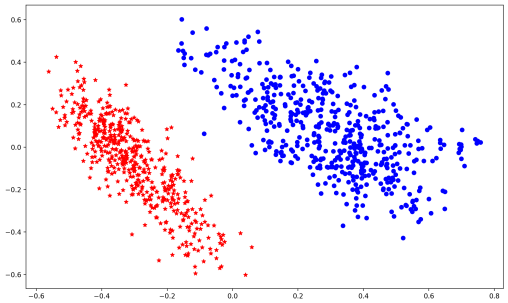

Figure 3: Clustering 1000 Most Biased Words (Original GloVe)

Male	**Female**
un	pregnant
hay	midwife
solo homers	breastfeeding
el	acrimony
solo homer	miscarriage

Table 6: Top 5 Most Biased Words (Original GloVe)

Direct Bias	0.157
Clustering Accuracy	0.988
Spearman Correlation for WS	0.5430
Google Analogies Accuracy	0.41

Table 7: Bias and Quality Evaluation (Original GloVe)

Word Pairs	Indirect Bias
receptionist, softball	0.3278
waitress, softball	0.1284
homemaker, softball	0.2460

Table 8: Indirect Bias (Original GloVe)

As we can see from our original data, there is a significant amount of direct bias compared to the Google news word2vec embeddings. This indicates that Twitter data is inherently more biased than news articles, which is what we would expect. However, the indirect bias for our selected pairs is significantly lower. We also see that the NLP task metrics are worse than the Google News word2vec ones. As with the Google News embeddings, we find that the most female-biased words exhibit stereotypical female traits. We notice, however,

that the male-biased words instead include masculine forms of foreign articles like the French "un" and Spanish "el", which we attribute to the high frequency of these words.

When we debias the GloVe embeddings, we again find that we are able to remove bias with respect to our measurements of direct and indirect bias, while also preserving their quality in regard to the word-similarity and analogy-solving tasks. Moreover, we see, on the other hand, that we are able to somewhat reobserve or reconstruct the hidden biases of these embeddings through the clustering of the 1000 most biased words as we obtained an accuracy of 72.6 percent. Our results after debiasing are shown in the tables and figure below.

Direct Bias	0.018
Clustering Accuracy	0.726
Spearman Correlation for WS	0.5420
Google Analogies Accuracy	0.47

Table 9: Bias and Quality Evaluation (Original GloVe)

Word Pairs	Indirect Bias
receptionist, softball	-1.9e-16
waitress, softball	-0.09
homemaker, softball	0.0

Table 10: Indirect Bias (Original GloVe)

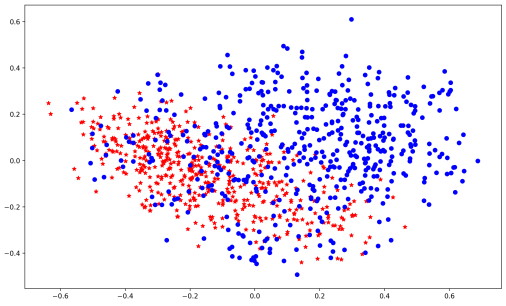

Figure 4: Clustering 1000 Most Biased Words[3] (Debiased GloVe)

4.3 Extension 2: Political Bias

We next explored political bias in word embeddings trained from media articles, with the following goals:

1. Do word embeddings trained from sources with a high perceived level of bias such as Breitbart News contain more direct and indirect bias?
2. Do debiased embeddings using the algorithm described in (Bolukbasi et al., 2016) differ significantly in performance?
3. Can we extend the algorithms described in (Bolukbasi et al., 2016) to debias political bias?

We used a news article corpus from Kaggle[4] that contained approximately 140K news articles from sources including the New York Times, Breitbart, CNN, Business Insider, the Atlantic, Fox News, Talking Points Memo, Buzzfeed News, National Review, New York Post, the Guardian, NPR, Reuters, Vox, and the Washington Post. We trained two word embeddings using Facebook's fastText with only the Breitbart articles (25K) and a random sample of approximately 25K articles from the full corpus. For this extension, we compre three word embeddings: the two we trained using fastText and the Google News embeddings used earlier for our baseline.

Similar to gender bias, we also need to create lists of definitional word pairs, equalize word pairs, and politically specific words. To create the definitional word pairs, we first used a thesaurus to extract words with similar meanings as "liberal" and "conservative". Then, we filtered out those word pairs that didn't have a dictionary definition that relates to politic. For example, (enlightened, ignorant) was filtered out during this step. Finally, we removed word pairs that have strong emotional undertones, such as (tolerant, intolerant). These steps ensures that these word pairs make sense when used in a political context and are not inherently strongly biased towards any ideology. A similar process was used to create the equalize word pairs, with the exception that we retained word pairs irrespective of their emotional undertones. We reason that a politically neutral word's embedding should not be affect by whether or not an equalize pair has emotional connotations, at least in the context of investigating political bias.

To create the list of politically specific words, we "crowdsourced" these words from the Internet using a large number of different politically motivated sources and then extracting political jargon from them. We specifically did not worry about

[4]https://www.kaggle.com/snapcrack/all-the-news

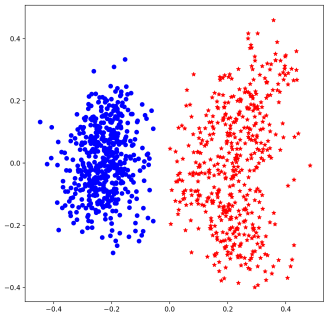

Figure 5: Clustering 1000 Most Biased Words (Biased GNews)

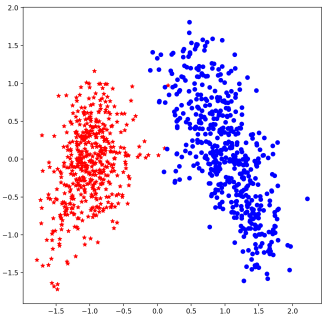

Figure 6: Clustering 1000 Most Biased Words (Biased Articles Sample)

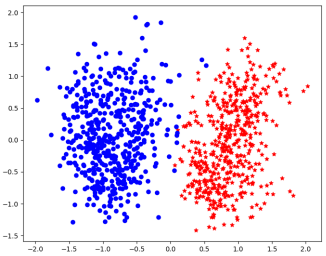

Figure 7: Clustering 1000 Most Biased Words (Biased Breitbart)

the degree of bias in those sources, because political jargon by nature can be very biased—in fact, the more biased the source, the more words we tended to be able to extract from that source. However, we did ensure that such jargon had an industry-standard meaning instead of just being a one-time "figure of speech" made up by the author.

It's important to note that we continued to investigate direct bias *with respect to* professions. For one, there appears to be a lot of systematic bias relating political affiliation and professions[5]. Secondly, we have no reason to believe that there is a much larger level of direct bias in any other observable human characteristic that exhibits the same level of "diversity". This eliminates characteristics such as gender, religion, and sexual orientation which may indeed have a larger level of direct political bias but do not have sufficiently many different values that each are referenced a nonzero number of times in our corpus.

We found the embeddings, before they were debiased, had the following results on the evaluation metrics described above in tables 11 and 12. We also include the clustering plots below.

Observe that the Google News embeddings exhibit a level of political bias similar to that of our baseline model on gender bias. This gives us confidence in continuing to use professions to assess direct bias in our embeddings.

Between the different embeddings, we see two main trends:

- Articles from major media sources appear more biased than *all* media sources[6], and perform more poorly on the word similarity and analogy tasks.
- Articles from Breitbart appear more biased than an aggregate of major media sources (including Breitbart), and perform more poorly on the word similarity and analogy tasks.

The clustering accuracy is very high for all three embeddings. This suggests that regardless of the source of the corpus (be it a balanced mixture of all political ideologies or just consisting of one ideology), there are still many words used in the corpus that can be distinctly associated with either liberal or conservative political jargon. The poorer performance on the word similarity and analogies tasks as we filter out more and more sources is a strong indicator that words are not used in the same contexts in the selected sources than they are in standard verbatim.

Interestingly, it appears that there is very little indirect bias in word pairs that are usually associated

[5]http://verdantlabs.com/politics_of_professions/

[6]the Google News corpus can be seen as a sample of all media articles, irrespective of the source's popularity

	Google News	Articles Sample	Breitbart
Direct Bias	0.0645	0.0741	0.0817
Clustering Accuracy	0.998	0.996	0.99
Spearman Correlation for WS	0.6857	0.5656	0.427
Google Analogies Accuracy	0.6666	0.5133	0.353

Table 11: Biased Embeddings Results

Word Pairs	Google News	Articles Sample	Breitbart
gay, atheist	0.0611	0.0101	0.0116
black, poor	0.0011	-0.0187	-0.0301
young, woman	0.0056	0.0038	-0.0028
white, rich	-0.0033	-0.0087	0.0068
old, man	-0.0309	0.0010	0.0049
straight, religious	-0.0052	0.0026	-0.0407

Table 12: Biased Embeddings Indirect Bias

with one political ideology. This may be due to the following:

- The word pairs we chose are not as strongly associated with a particular political ideology as previously thought. For example, whereas virtually all homemakers are women, not all women are liberal. It appears that such stereotypes about political ideologies are actually not statistically supported.
- We're not measuring indirect bias using the right set of word pairs. We need to come up with politically-neutral terms that are strongly associated with political ideologies. Not many such word pairs exist.

Despite the weak indirect bias, we can still observe that a very biased source (Breitbart) associates straight people with being religious, black people with being poor, and white people with being rich more strongly than the other media sources. On the other hand, Breitbart does not associate gender and age through political ideology as strongly as the other media sources.

After debiasing the embeddings, we found that they had the following results on the evaluation metrics described below in tables 13 and 14. We also include the clustering plots and the top 5 most biased words for both political ideologies before and after debiasing below.

The debiased embeddings indeed show a lower level of direct bias and clustering accuracy. In fact, the Breitbart embedding's cluster accuracy

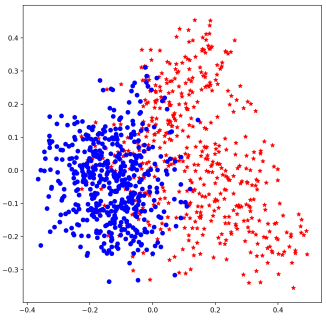

Figure 8: Clustering 1000 Most Biased Words (Debiased GNews)

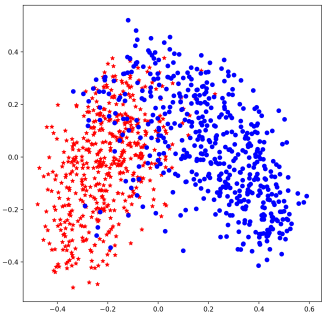

Figure 9: Clustering 1000 Most Biased Words (Debiased Articles Sample)

	Google News	Articles Sample	Breitbart
Direct Bias	0.00094	0.0017	0.0027
Clustering Accuracy	0.858	0.86	0.503
Spearman Correlation for WS	0.6826	0.5261	0.4277
Google Analogies Accuracy	0.6333	0.4667	0.4467

Table 13: Debiased Embeddings Results

Word Pairs	Google News	Articles Sample	Breitbart
gay, atheist	-0.0273	-0.0147	-0.0053
black, poor	0.0007	-0.0002	-0.0019
young, woman	$2.4934 \cdot 10^{-15}$	$-3.3637 \cdot 10^{-15}$	$7.6317 \cdot 10^{-16}$
white, rich	$-9.8606 \cdot 10^{-5}$	-0.0007	-0.0011
old, man	0.2844	-0.1952	-0.1905
straight, religious	$2.2081 \cdot 10^{-14}$	$-6.3628 \cdot 10^{-15}$	$1.6812 \cdot 10^{-14}$

Table 14: Debiased Embeddings Indirect Bias

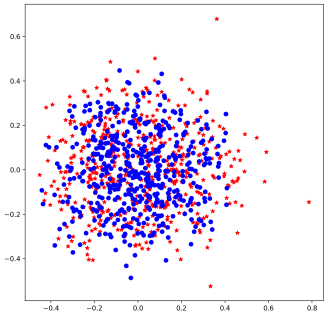

Figure 10: Clustering 1000 Most Biased Words (Debiased Breitbart)

"Left"		"Right"	
Biased	**Debiased**	**Biased**	**Debiased**
progressives	revolution	traditional	order
reactionary	old	conventional	new
liberals	past	rituals	changing
democrats	compromising	ritual	future
democrat	libertarian	customary	uncompromising

Table 15: Top 5 Most Biased Words (GNews)

"Left"		"Right"	
Biased	**Debiased**	**Biased**	**Debiased**
blasio	revolution	sushi	order
soros	redneck	juice	lefty
leftist	future	traditional	past
bernie	libertarian	recipes	communist
leftists	new	meals	old

Table 16: Top 5 Most Biased Words (Articles Sample)

has dropped to essentially 50%, which means the classifier essentially cannot tell the difference between the "most liberal" words and the "most conservative" words. This suggests that much fewer

"Left"		"Right"	
Biased	**Debiased**	**Biased**	**Debiased**
kathy	revolution	revolution	order
lena	communist	revolutions	libertarian
bias	future	revolutionary	past
regressive	old	sailors	new
kanye	redneck	principles	lefty

Table 17: Top 5 Most Biased Words (Breitbart)

words in the corpus can still be distinctly associated with either liberal or conservative political jargon. However, the debiased embeddings' performance on the word similarity and analogies tasks have generally failed to improve. This is probably due to the fact that most of the word pairs used in the word similarity task and the word tuples used in the analogy task are politically neutral, and thus any changes in their embeddings "cancels" out with each other.

With regards to indirect bias, we've managed to reduce the political association between race and wealth, young age and female sex, and heterosexuality and religious devotion. These trends apply across the board for all three embeddings. However, it appears that the indirect bias between homosexuality and atheism hasn't decreased, and the indirect bias between old age and male sex has drastically increased. These mixed results could suggest that we're not measuring indirect bias using the right set of word pairs, or that there is confounding factor at play. The words may be related politically through this other confounding factor, and our debiasing algorithm may have actu-

ally increased the effect of this confounding factor, hence giving us a worse indirect bias.

Finally, although the Breitbart embeddings originally appeared to be more biased than the other word embeddings, after we applied the debiasing algorithm, this no longer appears to be the case, both quantitatively and qualitatively. Qualitatively speaking, when looking at the top 5 most biased words for all three word embeddings before and after debiasing them, we see that while there are significant differences between the three embeddings before the debiasing, they appear much more similar after the debiasing. Words such as revolution, order, old, new, past, future, and libertarian appear in all three debiased embeddings' most biased list after the debiasing, whereas before the debiasing, the three word embeddings shared very few words in their most biased lists. This suggests that our debiasing approach is effective at making political bias uniform across various different media sources.

4.4 Extension 3: BERT

For our final extension, we sought to explore BERT embeddings, one of the more state-of-the-art contextualized word embeddings from Google. As BERT becomes more readily accessible, we predict that its use in industry and academia alike will see a surge in popularity.

We attempted to access pre-trained BERT embeddings, but found that querying for words in isolation resulted in misaligned word embeddings that were ultimately unusable. When querying for these embeddings, we verified their validity with respect to the word-similarity task. We found that the original embeddings only provided a Spearman correlation of around 0.11, which clearly indicates an issue with the manner in which we queried words in isolation (i.e. with no context). To further demonstrate the issues with the embeddings, we plotted the human-assigned similarity ratings against the cosine similarities from our word vectors which can be seen in Figure 11.

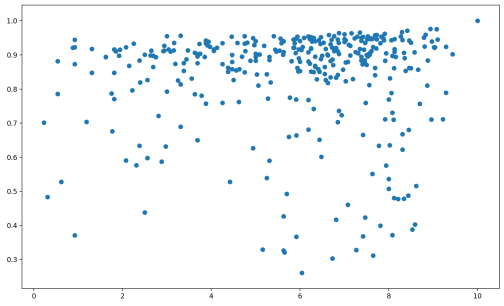

Figure 11: Word-Similarity Performance of BERT Embeddings

5 Conclusions

We examine bias present in various word embeddings used in NLP tasks. We reproduce the results in (Bolukbasi et al., 2016) and (Gonen and Goldberg, 2019) and also measuring their debiasing algorithm using evaluation metrics described in (Caliskan et al., 2017) and (Gonen and Goldberg, 2019). The results of our implementations closely follow the results reported in (Bolukbasi et al., 2016). In general, we find that there is significant indication of gender and political biases in existing embeddings and embeddings formed from articles of political news sources. Moreover, our chosen method of debiasing from (Bolukbasi et al., 2016) seems to demonstrate a strong ability to debias said embeddings.

However, there is still much work left to be done in this area. Gonen and Goldberg (2019) show that the work done in debiasing only superficially removes the debiasing, as the "bias ... is ingrained much more deeply in the embeddings space." They do so by clustering the most biased words in the original vocabulary using k-means, and find that the clusters still align with gender with a high accuracy. Thus, Gonen and Goldberg posit that existing metrics that have been explored in research may be good at measuring bias, but are not necessarily effective as a baseline for comparison for debiasing.

Acknowledgments

We would like to thank Professor Chris Callison-Burch for his guidance on our project, as well as Joao Sedoc, Reno Kriz, Anne Cocos, and Jishnu Renugopal for providing additional support and resources.

References

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. pages 4356–4364.

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

Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. 2018. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644.

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. *CoRR*, abs/1903.03862.

A Models

Code for this project can be found on Github, and the word embedding models can be found here.