Adversarial Approaches to Debiasing Word Embeddings

Stanford CS224N Custom Project
Mentor: Ethan Chi

Gordon Chi
Department of Computer Science
gsychi@stanford.edu

Benjamin Liu
Department of Computer Science
bencliu@stanford.edu

Abstract

In recent years, word embeddings have been ever more important in the world of natural language processing: techniques such as GloVe and Word2Vec have successfully mapped words to n-dimensional vectors that store precise semantic details and improve the quality of translation and generative language models. Since word embeddings are trained on human text, however, they also reflect unwanted gender and racial bias over decades of societal history. In this work, we propose that bias can be mitigated through the use of Generative Adversarial Networks. We experiment with two different problem formulations. First, we experiment with a discriminator that attempts to identify the gender bias of a vector, paired with a generator that minimizes the discriminator's performance on the task. Second, we experiment with a discriminator attempting to complete word analogies and identify the gender bias of the analogy, paired with a generator that only minimizes the discriminator's ability to identify the gender bias. Preliminary results on the WEAT scoring system show that both methods were successful in eliminating bias on commonly-used job words; qualitative analysis on similar words also show that racially or gender charged synonyms were considered less relevant to the debiased vector.

1 Introduction

Machine learning for natural language processing (NLP) leverages valuable data from human language for useful downstream applications such as machine translation and sentiment analysis. Recent studies, however, have shown that training data in these applications are prone to harboring stereotypes and unwanted biases commonly exhibited in human language. Since NLP systems are designed to understand novel associations within training data, they are similarly vulnerable to propagating these unwanted prejudices in downstream applications. A prominent example of this is seen in how Google translation services amplify career-based gender bias in translating gender pronouns between gender-neutral and gender-specific languages [1].

Gender bias is formally defined as the prejudice of one gender over another and constitutes the focal representative of bias in our study. Since word embeddings are leveraged as a fundamental component in training NLP systems both in tertiary and primary settings, unwanted biases displayed in these embeddings typically pose significant risk in propagating harmful prejudice in downstream applications . To see this concrete link, we offer an insight on word embeddings produced by training on the g2vNEWS corpus. When directly examining occupations most closely associated with the embeddings of "he" and "she," words such as "maestro, protege, and captain" emerge compared to "homemaker, nurse, and socialite," respectively. If these embeddings are further applied to text summary tasks in which rare words are substituted with word vectors of high cosine similarity, we can observe the potential for previously observed biases in our word embeddings to manifest. Concretely, a rare occupation-oriented word may be substituted with the representation for "homemaker" because it appears in female contexts.

In this work, we compare a pair of adversarial methods for debiasing word embeddings based on GANs, or generative adversarial networks [2]. Our first method is a simple generator-discriminator pair with a constrained generator loss, which incentivizes the generator to generate word embeddings that are close to their non-debiased counterparts. Our second method is a double discriminator based on FairGAN [3]; here, the generator is trained to generate embeddings such an embedding and its counterpart with an protected attribute reversed are indistinguishable by the discriminator. Our methods achieve strong performance on debiasing word embeddings as measured by the WEFAT metric. In particular, we find that our first method, despite being simple compared to prior work, performs competitively on a number of debiasing metrics.

2 Related Work

2.1 Word Embedding Debiasing

Research focused on debiasing word embeddings under the aforementioned metrics is in its primitive stages. Zhao et al. proposed a data augmentation technique deemed formally as "Counterfactual Data Augmentation" that attempts to debias word embeddings by modifying the embedding training corpus [4]. The proposed method involves swapping gender-specific words with their gender counterparts and appending this modified version of the corpus to the original corpus. When tested among pro-stereotypical and anti-stereotypical test sets, the augmentation method was found to lower the differences in F_1 scores between these sets significantly, indicating reduced gender bias in decision making.

Beyond this data augmentation technique, recent studies have focused on debiasing word embeddings through retraining techniques and complete removal of the gender subspace from word vectors. Bolukbasi et al., [5] for example, constructed a gender neutralization framework based on cosine similarity and orthogonal vector projections to remove gender bias as defined above from gender-neutral words. Using this method, the study was able to achieve lower gender bias as defined by the correlation measure described above with comparable performance in analogy-solving and coherence standard evaluation tasks between regular and debiased word embeddings of the w2vNEWS dataset. Some limitations of the mentioned studies include the infeasibility of data augmentation for gender-neutral languages and the subjectivity posed by less agreeable gendered words for use in gender subspace removal. Due to these limitations, researchers have also explored methods of retraining word embeddings under generalized fairness constraints outlined below.

2.2 Fairness Metrics

Three definitions of fairness and their respective limitations have been proposed for evaluating the relative fairness achieved in various machine learning systems by Hardt et al [6], including demographic parity, equality of odds, and equality of opportunity.

2.3 Generative Adversarial Debiasing

In 2016, Zhang et al. explored a variant to the classic generative adversarial network (GAN) model proposed by Goodfellow et al. to debias word embeddings and general feature embeddings through training [7]. By leveraging definitions of fairness, including demographic parity, equality of odds, and quality of opportunity proposed by Hardt et al., the study designed a GAN such that a discriminator was positioned to predict the protected attribute encoded in the bias of the original feature vector, while a competing generator was tasked with producing more debiased embeddings to compete with the discriminator. Adopting evaluations of bias outlined by Bolukbasi et al., the study demonstrated relative success in debiasing word embeddings across the aforementioned definitions of fairness with sustained performance in downstream analogy completion tasks.

Similar to work by Zhang et al., Xu et al. explored an adversarial approach (FairGAN) to debiasing feature embeddings in the form of enforcing statistical parity and thus, removing disparate treatment and disparate impact from continuous and discrete data [3]. The architecture of FairGAN features a generator that receives a noisy embedding paired with a protected attribute and is subsequently tasked with producing the original feature embedding of the example. Through this process, the generator competes with two discriminators, one of which identical to the original discriminator

design and the other aiming to predict the protected attribute associated with the designated feature embedding. Validating the efficacy of their debiasing approach through training linear classifiers and analyzing disparate treatment of samples through conditional probability analysis, the study provided strong evidence that FairGAN removes disparate treatment from feature embeddings and can generate synthetic, fair data.

3 Approach

3.1 Definitions

The gender bias associated with a single word is calculated by the Word Embedding Factual Association Test (WEFAT) [8]. Consider a word w and two sets of attribute words A and B. The WEFAT statistic associated with a single word w, denoted as $s(w, A, B)$, is defined as

$$s(w, A, B) = \frac{\texttt{mean}_{a \in A} \cos(\vec{w}, \vec{a}) - \texttt{mean}_{b \in B} \cos(\vec{w}, \vec{b})}{\texttt{std-dev}_{x \in A \cup B} \cos(\vec{w}, \vec{x})}$$

Observe that a positive WEFAT statistic for a single word suggests that a word is more positively associated with the words in the set A; a negative WEFAT statistic suggests that a word is more positively associated with the words in the set B. Defining A as a set of eight male terms (i.e. SON, HIM) and B as a set of eight corresponding female terms (i.e. DAUGHTER, HER), we use this as the metric to quantify the gender bias in a single word by our Glove Embeddings.

3.2 Generative Adversarial Model

To formalize our problem statement, we aim to develop a generative adversarial network that debiases 100-dimensional GloVe vectors trained on the Wikipedia dataset, using the WEFAT metric as a measurement of our success. For this goal, we experiment with two different problem formulations. [8].

3.2.1 Approach 1: Traditional Discriminator vs. Generator

Our model consists of a single discriminator and a single generator that compete to better understand the gender bias present in a vector. A discriminator $D(w)$ takes in a single normalized word embedding w and is designed to complete a binary classification task; it aims to identify the gender bias of a vector as determined by the WEFAT statistic. The paired generator $G(z, w)$ is given a 64-long vector z of noise coupled with the original word embedding w, and attempts to generate a fake, normalized word embedding $\tilde{w}$ that minimizes the discriminator's performance on the binary classification task.

For the loss function of our GAN, we consider a training pair $\{x, y\}$ and the corresponding fake outputs $\{\tilde{x}, \tilde{y}\}$ constructed by our generator, $y = \tilde{y}$. Then, given the discriminator's outputs $\hat{y}$ and $\hat{\tilde{y}}$ for the real and generated word embeddings, the discriminator's loss function is simply

$$D_{loss} = f(\hat{y}, y) + f(\hat{\tilde{y}}, \tilde{y})$$

where f is defined a Binary Cross Entropy Loss. The Generator's Loss Function on the other hand is defined as

$$G_{loss} = \alpha \cdot g(\tilde{X}, X) + f(\hat{\tilde{y}}, 1 - \tilde{y})$$

where g is defined as Mean Squared Error loss, and α is a tuneable constant. The first part of the equation penalizes the generator for making word embeddings too far from the original equation, whereas the second part rewards our generator for successfully tricking the discriminator. α is set to 10^{-6} during our last meaningful runs; it turned out that the generator learned to construct vectors close to the initial embedding even when its error only takes up a small part of our final loss function.

The architecture of both the discriminator and generator are both similar to a traditional feed-forward network, as seen in Figure 1. More details of the network are shown in Table 1 below.

Model	**Layer Type**	Neurons	Activation
Discriminator	Input	100	None
	Hidden	1024	LeakyReLU
	Hidden	1024	LeakyReLU
	Hidden	512	LeakyReLU
	Hidden	256	LeakyReLU
	Output	1	Sigmoid
Generator	Input (Noise)	64	None
	Hidden	100	LeakyReLU
	Input (Embedding)	100	None
	Hidden	1024	LeakyReLU
	Hidden	1024	LeakyReLU
	Hidden	512	LeakyReLU
	Output	100	None, output normalized

Table 1: **Network Architecture for the generator and discriminator**. Note that the generator has two input layers: a noise input of length 64 is first transformed into an output of length 100 before it is concatenated into the 100-long word embedding to create a size-200 vector. This size-200 vector is then passed into the remaining hidden layers and results in an output word embedding of length 100.

3.2.2 Approach 2: FairGAN Double Discriminator vs. Generator

As described earlier, the FairGAN paper [3] introduces the use of a protected attribute in Generative Adversarial Network (GAN) models: a model is given an input X and maximizes its ability to predict a desired output y while minimizing its understanding of a protected attribute S given input X. We adopt this framework in the context of word analogies. Here, $X = (x_1, x_2, x_3) \in \mathbb{R}^{300}$ is a concatenation of the GloVe embeddings of three words in a 4-word-analogy, $y \in \mathbb{R}^{100}$ is the GloVe embedding of the fourth, and $S \in \{0, 1\}$ is a protected binary value that determines whether the fourth word in the analogy has a female or male bias present in its representation.

From an architecture perspective, the generator (G) is based on the autoencoder model referenced from the original FairGAN paper; however, instead of receiving a direct noisy embedding appended to S, our generator takes in a noisy embedding $n \in \mathbb{R}^{100}$ concatenated with input X and S. Generated outputs are fed alongside original embeddings to two separate discriminators in varying formats. Let D_1 denote the first discriminator that aims to distinguish between real and fake embeddings, and let D_2 denote the second discriminator that aims to successfully predict to protected attribute associated with the sample. D_1 received as input both the fake and real embeddings labeled as real, while D_2 receives a double set of fake and real embeddings. Specifically, in training D_2, the original embeddings are copied such that their protected attribute labels are inverted. This double set of embeddings is then fed to the generator to produce two corresponding sets of fake embeddings, which altogether are involved in training D_2 and G. The training procedure and associated loss functions for this model are identical to those described in the original FairGAN paper barring the variations to the input word embeddings and protected labels described above. A detailed description of our network structure can be seen in Table 2.

3.3 Evaluation Metrics

We evaluate the performance of our model by the WEAT score, as well as analyzing the top 10 most similar words determined by cosine similarity scores.

3.3.1 Word Embedding Association Test

First introduced by Caliskan et. al in 2017 [8], the WEAT associated is formally defined as a define X, Y as two sets of target words of equal size, and A, B as two sets of attribute words. Then, the WEAT score $s(X, Y, A, B)$ is defined as

Model	**Layer Type**	Neurons	Activation
Discriminator 1 and 2	Input	300	None
	Hidden	512	ReLU
	Hidden true	256	ReLU
	Hidden	128	ReLU
	Output	1	Sigmoid
Generator	Input	401	None
	Hidden	512	Tanh
	Output	300	Sigmoid

Table 2: **Network Architecture for the FairGAN-based generator and dual-discriminator model**. Note that the generator receives the length-300 3-word embedding concatenated with a length-100 noise vector and a length-1 protected attribute label. This structure is consistent with that presented in the original FairGAN paper with slight modifications to accommodate word vector inputs.

$$s(X, Y, A, B) = \sum_{x \in X} s(x, A, B) - \sum_{y \in Y} s(y, A, B)$$

where

$$s(w, A, B) = \texttt{mean}_{a \in A} \cos(\vec{w}, \vec{a}) - \texttt{mean}_{b \in B} \cos(\vec{w}, \vec{b})$$

Note that since cosine similarity is bounded between -1 and 1, our WEAT scores range from a value of -2 to 2. Defining A as a set of 8 male words and B as a set of 8 female words, we have that a positive score denotes a heavy male bias for words in X over words in Y, whereas a negative score denotes a heavy female bias for words in X over words in Y.

3.3.2 Cosine Similarity Analysis

For GloVe Embeddings, we choose a few unseen and/or common words to analyze the ten most similar words prior to debiasing. We then take an aggregate debiased vector (i.e. the average output $G(z, w)$ for a word w over 100 different noise vectors z) and calculate the new cosine similarity scores for those 10 words. We then provide qualitative analysis of the changes and attempt to decipher the positive or negative changes presented by our model.

4 Experiments

We run experiments on both Approach 1 and 2.

4.1 Data

4.1.1 Approach 1

We train our model on a list of 177 job-related words and their corresponding GloVe embeddings; larger training sets did not yield better results. The embeddings are trained on the Wikipedia dataset; each word embedding is of dimension 100. The exact list of words can be found in the Appendix.

4.1.2 Approach 2

We utilize the Google analogy data set by Mikolov et al. [9], and keep all 493 analogies containing gender-related words. Again, the full list of words can be found in the Appendix. To prepare the dataset, we separated the first three words of the analogy, and concatenated their corresponding word embeddings from the "glove-wiki-gigaword-100" model into a 300-length embedding. The glove embedding of the fourth analogy component was deemed the label of the example, and the WEFAT score of this fourth word was deemed the protected attribute label for the entire example.

4.2 Experimental details

4.2.1 Approach 1

Both the predictor and the generator were trained using the Adam optimizer [10] for 5000 epochs with a batch size of 128 and the constants $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The learning rate of the generator was 10^{-3}, whereas the learning rate of the discriminator was 10^{-4}.

4.2.2 Approach 2

The generator was pretrained for 1000 epochs with a batch size of 64 on the original 300-length embeddings in the training set. Following this, the complete generator, dual-discriminator model was trained for 2000 epochs with Adam optimization, under the constants $\beta_1 = 0.5$ and $\beta_2 = 0.999$. Consistent with the original FairGAN paper, the learning rate for the generator and discriminators was 10^{-3}.

4.3 Results

4.3.1 Approach 1

The loss values of our training curve are shown in the Figures below. After training, embeddings in our training set saw a decrease in the absolute value of its WEFAT statistic by an average of 0.035.

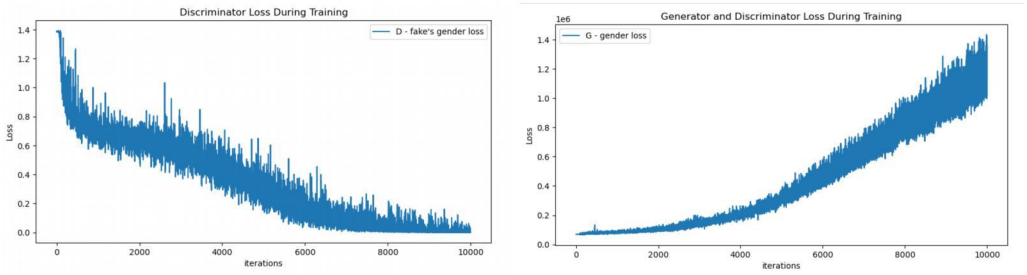

Figure 1: **Training loss over time**. Observe that iterations is incremented each time we pass through a batch. There are two iterations per epoch, which results in the x-axis that spans from 0 to 10,000.

To better quantify our results on the training set and never-before-seen words, we performed qualitative analysis by computing WEAT scores.

Set A	**Set B**	WEAT (Initial)	WEAT (After)
farmer, janitor, manager, mathematician	maid, dancer nanny, artist	0.552	0.464
boss, leader, manager, executive	nurse, teacher nanny, artist	0.880	0.783
mathematician, lecturer, doctor, executive	maid, dancer policewoman, actress	0.568	0.464
novelist, poet, historian, writer, official, editor	dancer, artist, dance, performer, actress, ballerina	0.707	0.640

On a mixture of seen and unseen job words, as shown above, our model does a decent job of debiasing words, i.e. the words in set A and set B are less associated with a gender attribute after training a generator/discriminator to account for this bias. While testing our results on a test set is less meaningful for this task (given that our job dataset contains the majority of professions that come up in conversation), we also show that our model does well on words it has yet to see:

Set A	Set B	WEAT (Initial)	WEAT (After)
leader, executive, maestro, financier, pilot, carpenter	policewoman, actress, housekeeper nanny, executive, homemaker	1.148	0.947

To further demonstrate the improvement and how our word embeddings have changed through our training process, we analyze cosine similarity of the top ten most similar words for a few words in our training set, as well as the test set:

Rank (Initial)	Word	WEFAT (Initial)	WEFAT (After)	Change
1	doctor	0.752	0.785	0.033
2	nurses	0.740	0.708	-0.032
3	physician	0.688	0.719	0.031
4	nursing	0.688	0.687	-0.001
5	dentist	0.680	0.767	0.077
6	therapist	0.673	0.681	0.008
7	midwife	0.669	0.687	-0.081
8	hospital	0.665	0.682	0.017
9	surgeon	0.659	0.687	0.028
10	psychiatrist	0.649	0.679	0.030

Table 3: Top 10 most similar words for `nurse` according to our GloVe Embeddings, and the cosine similarity scores after our debiasing in Approach 1. Note that the only word that has a large decrease in cosine similarity is midwife, which has a clear gender connotation. Other words such as doctor, dentist, surgeon and psychiatrist are considered more similar to `nurse` due to their connection to the field of healthcare, which is what we desire.

Rank (Initial)	Word	WEFAT (Initial)	WEFAT (After)	Change
1	housekeeper	0.713	0.755	0.042
2	filipina	0.627	0.486	-0.142
3	flor	0.607	0.452	-0.155
4	maids	0.602	0.543	-0.059
5	servant	0.590	0.615	0.025

Table 4: Top 5 most similar words for `maid` according to our GloVe Embeddings, and the cosine similarity scores after our debiasing in Approach 1. Note that though we do not train our network to debias other traits, i.e. racial bias, our model was still able to dampen the connection between filipina and maid. The same can be seen for the word flor, which has a large decrease in cosine similarity.

A total of analysis on three more words – homemaker, author, dancer – have similar results and can be seen in the attached source code.

4.3.2 Approach 2

The training progressions over our model components and pretrained generator are shown in A.4, Fig. 2. From our initial observations of the loss progression throughout training, it appeared that the model failed to properly converge. This hypothesis was confirmed by further experiments analyzing WEFAT score changes and individual words. Overall, generated word embeddings displayed an average cosine similarity of 0.689 compared to original word embeddings. In addition, the debiasing capabilities of our FairGAN variant were heavily inconsistent, as seen in A.4, Fig. 3. To further evaluate these inconsistencies on the basis of perceived debiasing potential, we performed qualitative analysis on the words with the highest WEFAT score differences. Specifically, we analyzed differences in cosine similarity between the original and generated word embeddings, comparing the the original embedding's 10 most similar counterparts. One example of this analysis is seen in A.3.

5 Analysis

5.1 Approach 1

Our model benefits greatly from a smaller scope of training. Even when we only consider the task of debiasing gender, a more diluted dataset of 1200 words, including many of the top 200 most commonly used adjectives and nouns, did not yield strong results for the word sets containing just jobs or just adjectives. This suggests that domain knowledge is important; debiasing is only successful if our GAN is given a specific type of noun/adjective and a specific form of bias to remove.

Moreover, while many of the cosine similarity scores do change in accordance to the bias we observe, it seems that the plural form of word embeddings always end up with a lower cosine similarity than desired. This may prompt us to look into adding plural forms of all jobs into our dataset, and observing whether we can achieve higher performance overall.

5.2 Approach 2

Results from our qualitative analysis of debiased word vectors and WEFAT score comparisons reveal that the FairGAN model is largely ineffective in debiasing word vectors. This conclusion was further supported by analyzing cosine similarity scores among original and debiased vectors, which yielded inconsistent and largely unpredictable results.

One possible reason for these inconsistencies is that the original FairGAN paper is tailored to debiasing feature vectors rather than word embeddings. Thus, the proposed loss functions may not be properly tailored to our desired debiasing task as a result. In addition, both studies by Zhang et al. and Xu et al. [3, 7] indicate that the proposed adversarial designs are inherently difficult to train and require precise hyperparameter tuning methods to converge the model in training. The adaptation of the FairGAN model to word embeddings may have rendered the suggested training parameters as largely inapplicable, potentially resulting in the training diffulties that we observed. Thus, future error analysis must be focused in these areas.

6 Conclusion

Ultimately, in this work, we demonstrate a relatively unexplored approach – Generative Adversarial Networks in the field of NLP – and show that it has potential in training unbiased machine learning models. With two different problem formulations, we show that a generator is able to identify the reasons for bias present in a small sample of data and create improved embeddings that mitigate the stereotypes at hand.

6.1 Limitations

While this project shows promise, there are a few important areas for improvement. First, as mentioned previously, our task has a hard time generalizing and only performs well on a smaller domain of words. This may be due to either the complexity of the task (i.e. having to accurately predict a 100-dimensional vector from a 300-dimensional input as seen in Approach 2), or simply due to how difficult it is to construct a large dataset that contains only word embeddings with problematic semantics.

Additionally, we observe that when changing the weights of the loss function, our model must sacrifice average cosine similarity to improve the debiasing of word embeddings and vice versa. While this may be due to the fact that network architecture and hyperparameters may not be optimized, we believe that this may also be a limitation of our approach in general.

Future work will look into tweaking network architecture and hyperparameter tuning, as well as seeing whether our approach attains similar/better performance across different forms of bias and languages as well.

References

[1] Marcelo OR Prates, Pedro H Avelar, and Luis C Lamb. Assessing gender bias in machine translation: a case study with google translate. *Neural Computing and Applications*, pages 1–19, 2019.

[2] Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

[3] Depeng Xu, Shuhan Yuan, Lu Zhang, and Xintao Wu. Fairgan: Fairness-aware generative adversarial networks. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 570–575. IEEE, 2018.

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

[5] Tolga Bolukbasi, Kai-Wei Chang, James Zou, Venkatesh Saligrama, and Adam Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, page 4356–4364, Red Hook, NY, USA, 2016. Curran Associates Inc.

[6] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. *CoRR*, abs/1610.02413, 2016.

[7] Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 335–340, 2018.

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

[9] Tomás Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. *CoRR*, abs/1310.4546, 2013.

[10] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

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

A Appendix

A.1 Measures of Gender Bias

Two prominent methods have been proposed for measuring gender bias in word embeddings: gender subspace analysis and word embedding factual association test (WEFAT). In 2016, Bolukbasi et al. analyzed gender bias in word embeddings by proposing a correlation measure between the projection of a candidate word embedding onto a gender subspace and the candidate word's bias measured by human ratings [11]. To construct their referenced gender subspace, they categorized a series of common gender-specific and gender-neutral words using a support vector linear machine and aggregated gender pairs. Leveraging these pairs, they used principal component analysis (PCA) to identify the maximally variant eigenvector used for gender direction labeling. This method demonstrated results consistent with crowd-worker evaluation benchmarks. Another prominent metric for measuring gender bias in word embeddings is the WEFAT proposed by Caliskan et al. Inspired by the Implicit Association Test, the WEFAT considers whether given target word embeddings carry knowledge of a given property such as gender or race. In essence, the ability of a systematic classifier to predict the property given the word embedding is measured [8].

A.2 Training Data

Words used to train GAN in Approach 1 (Identifying Gender Bias Experiment): changer, librarian, drafter, prosthetist, anthropologist, caster, scout, anesthesiologist, actuary, tailor, farmer, agent, optometrist, engineer, archeologist, hydrologist, laborer, pediatrician, surgeon, doctor, educator, host, etcher, judge, teacher, patternmaker, computer, finisher, chemist, sewer, child, auditor, dentist, geoscientist, educational, writer, interpreter, baker, reporter, fitter, demonstrator, usher, jailer, investigator, cutter, farming, grinding, audiologist, mechanic, tuner, busines, painter, cook, slaughterer, manager, clerk, rigger, barber, repairer, author, runner, analyst, coating, hunter, microbiologist, chiropractor, choreographer, receptionist, presser, waitress, court, curator, sorter, bookkeeping, jeweler, painting, nurse, tender, obstetrician, entertainer, taper, pruner, interpretor, crushing, scaler, editor, welding, bellhop, architect, cutting, cleaning, courier, teller, epidemiologist, singer, adjuster, developer, shipping, designer, molder, lawyer, electrician, lecturer, typist, producer, biophysicist, firefighter, psychiatrist, plumber, property, faller, photographer, musician, director, engraver, cashier, bailiff, actor, artist, interviewer, umpire, glazier, clinical, sociologist, cabinetmaker, surveyor, compensation, optician, proofreader, technologist, nutritionist, orthodontist, welder, cleaner, hairdresser, embalmer, grader, buyer, boss, hostess, trimmer, veterinarian, dancer, telemarketer, packer, clergy, forester, athlete, gynecologist, area, roofer, mathematician, butcher, animator, maid, coin, psychologist, pourer, trapper, correspondent, janitor, sailor, waiter, composer, fisher, chef, atmospheric, boilermaker, captain, programmer, escort, conservator, packager, researcher, statistician, cartographer, tiler

Words used to filter dataset for Approach 2 (FairGAN Analogy Completion Experiment): he, him, father, son, boy, brother, brothers, dad, groom, husband, king, man, policeman, prince, sons, woman, she, men, women

A.3 FairGAN Qualitative Analysis

Rank (Initial)	**Word**	Cosine Similarity (CS) (Initial)	CS (After)	Change
1	sons	0.797	-0.012	-0.809
2	brother	0.687	0.081	-0.606
3	sisters	0.672	0.004	-0.677
4	cousins	0.652	0.225	-0.426
5	siblings	0.626	0.045	-0.558

Table 5: Top 5 most similar words for `brothers` according to our GloVe Embeddings, and the cosine similarity scores after our debiasing. Note that cosine similarity decreased unexpecetedely and somewhat unpredictably among gender parallel words such as "sons" and "brother" which is undesired.

A.4 FairGAN Training and Debiasing Results

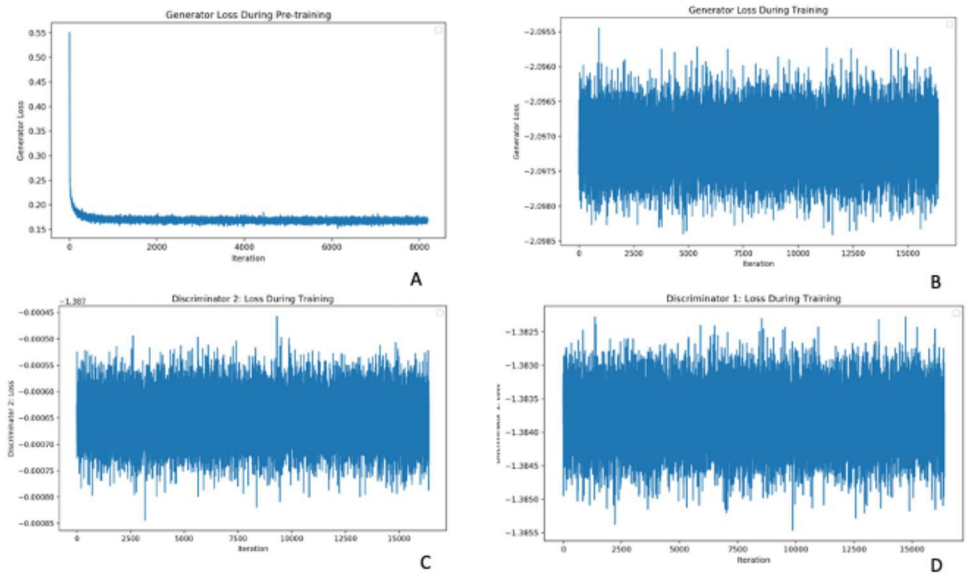

Figure 2: **Training loss over time**. Observe that iterations is incremented each time we pass through a batch. There are six iterations per epoch, which results in the x-axis that spans from 0 to 15,000. We can also see that although the generator trains as expected during pretraining, during regular training, the GAN displays abnormal training patterns. This is a indication that the model may have struggled to converge properly.

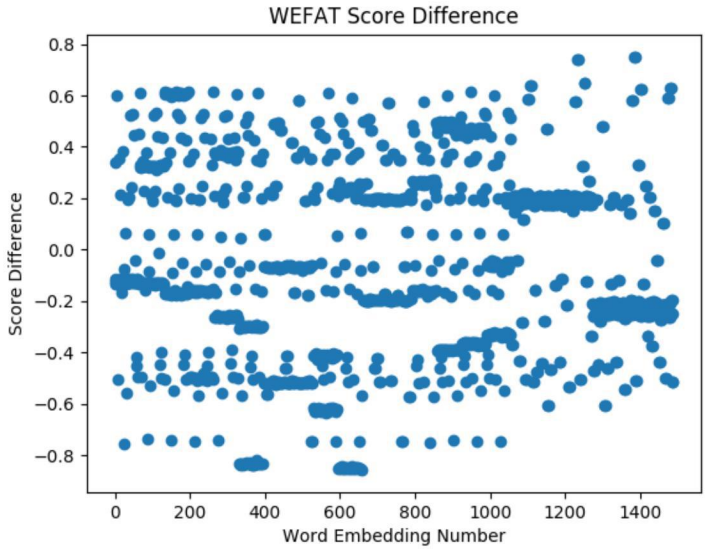

Figure 3: **FairGAN Model: WEFAT Difference Scatter Plot**. This plot captures differences between the WEFAT scores of old and new word embeddings produced by the trained generator; scores about 0 indicate evidence of gender debiasing while scores below 0 indicate the alternative. As seen, the model demonstrates relatively inconsistent results across 1400 test word embeddings.