Algorithmic Bias in NLP

Kate Isaksen, Izzi Grasso, David Russell

May, 2020

1 Algorithmic Bias

Definitionally, when an algorithm's use or output results in unfairness, that algorithm is biased [9]. In the midst of the "era" of big data, algorithms are being increasingly used to make decisions, make predictions, and gain understanding about people and the world [3]. As the creators and designers of algorithms, this gives technologists increasing power in all aspects of life; hiring, loan applications, criminal justice, health care, transportation, education, etc. Many of the decisions for which algorithms are used are highly regulated. Using hiring as an example, it is illegal for hiring decisions to be discriminatory, and there are specific attributes that are protected; age, race, gender identity, sexual orientation, religion, ethnicity, and socioeconomic status. If a job were to be offered, or not offered, based on any of these attributes that would be illegal and discriminatory. The law also prohibits discrimination by algorithms when being utilized in these critical decision-making landscapes [9].

However, whether the decisions are made by humans, by algorithms, or anything in between, determining whether or not a decision is biased, or discriminatory, can be difficult. The underlying reasons behind human decisions are often not available to outsiders, and even the decision-maker themselves may not be fully aware of all of the factors involved [9]. For example, a study by Danziger et. al. exposed that a defendant is two to six time more likely to be released if they are one of the first three prisoners after a food break as opposed the last three prisoners considered [4]. The finding gives evidence that regardless of training, values, or intellect, there is a strong psychological effect associated with providing rest and increasing glucose levels in the body. These results were quite shocking given that the assumption and requirement of judges is that rulings are based solely on laws, facts, and evidence, not biases, mood, or hunger. Furthermore, recent sociological research suggests that racial and gender bias are still rampant. A 2018 study showed that male instructors administering an identical online course as a female instructor received higher teaching evaluations, even for questions that are not instructor specific [14]. At Apple only 6 percent of their technical workers are black, even though black people make up 13 percent of the US population and 10 percent of computer science degrees earned in 2018 [7].

Many technologists claim that automating these decisions will fix this problem of bias. Algorithms won't get cranky and hungry and dole out unfair outcomes in court. Algorithms are not sexist. Algorithms are not racist. But algorithms are greedy. They are greedy in the sense that many algorithms, especially the increasingly popular machine learning algorithms, require a lot of data. For this paper we will focus on machine learning algorithms, and natural language processing specifically. In its simplest form, a machine learning algorithm is an algorithm that learns from data. So while it is true that an algorithm is not racist or sexist or homophobic, it is also true that racism, sexism, and homophobia can be found in data and therefore learned by algorithms. In 2014 Amazon created a team to build a system that would sort resumes. The general idea was given say one hundred resumes, this system could pick the top five candidates to hand to recruiters to individually interview. The models were trained on the hiring patterns of Amazon over a ten year period. Initial results showed that any resume with the word "women's" or "women" on it, or say "captain of the women's chess club" was downgraded. It also downgraded graduates of all-women's colleges. It was also found that the technology favored candidates using terms more commonly found on men's resume's such as "executed" and "captured" [5].

Is this discriminatory? Due to the opaque nature of the systems, it is not clear what the underlying mechanisms of the decisions made by the algorithm. The outputs can be analyzed, which is how they came to understand that women's resumes were downgraded more often than men's resumes, but it is not clear that it was because they were women's. Recall that the definition of a biased algorithm is based on fairness, not discrimination. But what is fair?

2 Fairness

Fairness is a heavily debated idea among philosophers and technologists when it comes to algorithms, and there are many incompatible ways to measure fair outcomes for machine learning algorithms; demographic parity, positive predictive parity, negative predictive parity, false positive parity, equality of opportunity, to name a few [10]. Derek Leben uses the famous example of COMPAS, the risk assessment algorithm that was built for a criminal justice setting to determine the risk of prisoners reoffending. It has been used for sentencing, parole, and other decisions in the criminal justice process. Is it fair? It depends. Broken down by a few normative principles of fairness here are the results:

Positive predictive: Fair. It correctly predicts reoffense at the same rate for both groups.

Demographic parity: Fair. 50% of both black and white prisoners are labeled "high risk".

False positive: Unfair. 16% of black prisoners were incorrectly labeled as high risk, while only 9.6% of white prisoners were labeled as high risk.

Equality of Opportunity: Unfair. 93% of white prisoners who were dangerous were correctly labeled, while only 64% of black dangerous prisoners were

labeled correctly.
So is it fair?

The reason why this question is so difficult to answer is fair means different things to different people, and it also changes with context and domain. Researcher Arvind Narayanan claims that trying to find one true definition of fairness in computer science is a wild goose chase. Because of the ubiquity of automation and software, what is fair in one domain, say criminal justice, may not be fair for another, like college admissions. It is important that algorithms not only uphold a definition of fairness, but that uphold human values in the domain in which they are deployed [**cit˙9**]. The reason why Propublica declared COMPAS was unfair was because although the accuracy was the same for white prisoners and for black prisoners, the definition of fairness in this domain is innocent until proven guilty. This means that the most egregious error COMPAS could make is to falsely classify a prisoner as high risk when in fact they are not, i.e. false positives. Because there lacks parity of false positives–black prisoners were more often falsely labeled as high risk than white prisoners–COMPAS can be reasonably defined as unfair and therefore biased racially.

Unfortunately, the question of fairness is often unclear. Amazon's resume sorting algorithm is a prime example of this. Regulations against discrimination in the area of hiring follow process fairness. The decision cannot be made based on a protected attribute. Oftentimes companies also have policies to incentivize outcome fairness, e.g. they have a goal of having equal numbers of male and female technical workers at their company. Amazon's hiring algorithm was eventually tossed because under the definition of outcome fairness, it was biased against women. However, it is essentially impossible to know whether or not it followed process fairness, because the underlying mechanisms of the models are not interpretable to humans. It then becomes incredibly difficult to build an algorithm that is both computationally powerful enough to process text, but also transparent enough to uphold the values of this particular domain, process fairness.

3 Natural Language Processing

Natural Language Processing (NLP) refers to the way computers understand, interpret, and manipulate human language. "NLP allows machines to understand and extract patterns from such text data by applying various techniques such as text similarity, information retrieval, document classification, entity extraction, clustering" [8]. Text similarity is one of the main parts of NLP and is used to find the closeness between two texts by its meaning. The first step of text similarity is text preprocessing, where it transforms the text into a more digestible form so that the learning algorithms can perform better. This includes splitting paragraphs into sentences, converting all letters to lowercase, removing special characters, converting number words to numeric form, and extra whitespaces. Other techniques include word embedding and vector similarity.

4 Embeddings

Word embeddings allow us to represent words in the form of N-dimensional vectors. Each word is mapped onto a series of zeros and a one, with the location of the one corresponding to the index of the word in the vocabulary. These vectors should "capture the meaning of words, the relationship between words, and the context of different words as they are used naturally" [12]. Some key characteristics used when setting word embeddings are:

- Every word has its own unique word embedding.
- The word embeddings are multidimensional.
- The embedding should capture the "meaning" of the word.
- Similar words should have similar embeddings.

The simplest word embedding scheme is One-hot encoding. With this encoding, "the embedding space has the same number of dimensions as the number of words in the vocabulary. Each word embedding is predominately made up of zeros, with a '1' in the corresponding dimension for the word"
[12]. This embedding scheme is very simple so it has some problems that arise including that the number of dimensions increases linearly as more words are added to the vocabulary, the embedding matrix is mostly made up of zeros, and that there is no shared information between words and similar words.

One of the most popular algorithms for word embeddings is the Word2Vec algorithm [13] from Google. "It is proven useful since it is high quality, publicly available, and easy to incorporate into any application" [1]. It takes a text corpus as input and produces the word vectors as output. It creates a training text data and then learns vector representations of words. The Continuous Bag-of-Words (CBOW) model and Continuous Skip-Gram Model are two parts of the Word2Vec approach to learn the word embeddings. "The CBOW model learns the embedding by predicting the current word based on its content. The Continuous Skip-Gram model learns by predicting the surrounding words given a current word" [2].

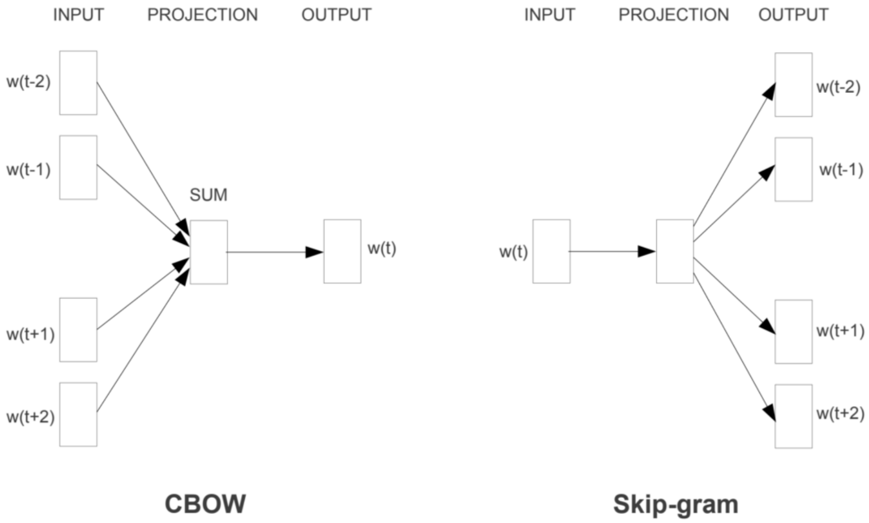


Figure 1: Word2Vec Training Models

4.1 Vector Similarity

Since similar words have similar embeddings, this means that the similar words have vectors that are close together. “The vector differences between words in embeddings have been shown to represent relationships between words” [1]. The cosine similarity of words is the measure between two non-zero vectors of an inner product space that measures the cosine of the angle between them. Two vectors with the same orientation have a cosine similarity of 1, as seen in the first example of Figure 2 with France and Italy. They are similar because they are both European countries. Two vectors oriented at 90°relative to each other have a similarity of 0, shown in the second example of ball and crocodile. These two words are not similar. Two vectors that are diametrically opposed have a similarity of -1. The figure shows an example of Rome-Italy and France-Paris. These vectors are similar but in opposite orders.

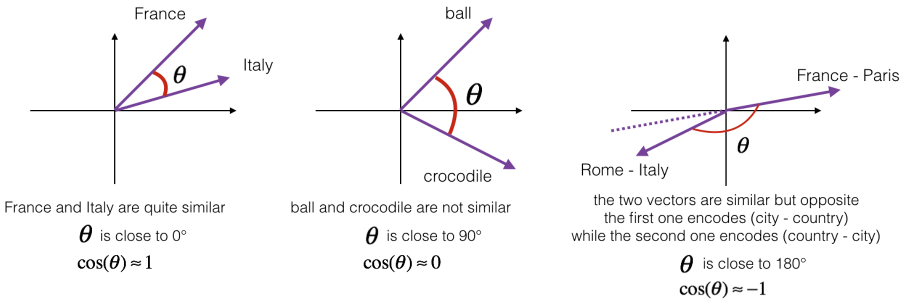

France
Italy
θ
France and Italy are quite similar
θ is close to 0°
cos(θ) ≈ 1
ball
crocodile
θ
ball and crocodile are not similar
θ is close to 90°
cos(θ) ≈ 0
France - Paris
Rome - Italy
θ
the two vectors are similar but opposite
the first one encodes (city - country)
while the second one encodes (country - city)
θ is close to 180°
cos(θ) ≈ −1

Figure 2: The cosine of the angle between two vectors is a measure of how similar they are

4.2 Properties of Embeddings

Word embeddings have many properties including recognizing words that are similar and identifying the linguistic relationship of words. The similarity property of word embeddings allows applications to work with the words that have been misspelled and words that have not been seen before. “Similarity” in this sense is defined as the cosine similarity. The linguistic relationship is the linear relationship between vectors discovered from the use of language in the training set. For example, “the transformation between the vector for ‘man’ and ‘woman’ is similar to the transformation between ‘king’ and ‘queen’, ‘uncle’ and ‘aunt’, ‘actor’ and ‘actress’, generally defining a vector for ‘gender’” [12]. The linguistic relationship allows words to be added and subtracted and lets us solve word analogies of the form A is to B as C is to X. Looking at the gender relationship, we can expect the vector differences of man: woman, king: queen, and brother: sister to be all roughly equal, so when given man is to woman as king is to X, we can determine that the X would be queen.

4.3 Applications for Embeddings

Word embeddings can be used for more than just finding similarities between specific words and phrases. They can be used for Google searches, resume sorting, document retrieval, span detection, music recommendation systems, analyzing survey responses, and many more. Word vectors are used with Google searches to improve the accuracy of searches and find things that are misspelled or aren’t exactly what is searched.

Word embeddings can also be used to uncover sexual harassment patterns. In the paper *Uncover Sexual Harassment Patterns from Personal Stories by Joint Key Element Extraction and Categorization* [11], online sexual harassment reports were studied and labeled with categorizations of location, time, type of harasser, age of harasser, and single/multiple harasser(s). It was found that harassment occurred more frequently during the night time than the day time, the majority of young perpetrators engaged in harassment behaviors on the streets, young harassers are more likely to engage in verbal harassment rather than physical harassment, and adult perpetrators are more likely to act alone and on public transportation.

5 Word Embeddings and Social Bias

Word embeddings are useful because they capture features about how language is used. Unfortunately, when language is used in ways which express societal biases, such as sexism or racism, the embedding is likely to have these characteristics as well. This is troubling because embeddings are used as a first "pre-processing" step for a large fraction of text-based NLP algorithms [6]. As the old adage says—"garbage in, garbage out"—if the embeddings are biased, the downstream task will also have elements of this bias.

A pioneering work in discovering and quantifying this bias was the 2016 paper Man is to *Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings* [1]. The primary thesis is that words which should not have a gender associated with them, such as names of professions, in fact have severely stereotypical gender biases. Prior to this paper, it had been well established that word embeddings can be used to effectively solve analogies, such as "man is to woman as king is to x?" In this case, queen completes the analogy because the difference between the embeddings of man and woman is most similar to the difference between king and queen. This is reasonable because king and queen, by definition, differ by gender. The troubling reason for the title of the paper is that the solution to "man is to woman as computer programer is to x?", is *homemaker.* There is not a definitional gender difference between computer programer and homemaker so this suggests that this association is caused by male and female stereotypes associated with these roles. This means that societal biases could be magnified since women would be less likely to be associated with computer programming in search results and resume selection.

The authors developed an automated process to produce these analogies and then presented them to crowd-workers for review. They experimented with ten pairs of seed words, such as *she-he*, *her-him*, *woman-man*, which defined a direction which the analogy would complete. All of these differences are relatively similar but due to different contexts, they vary slightly. For a given pair of seed words, for example man-woman, they sampled a random word x from the set of all other words. Then they sought to complete the analogy "man is to woman as x is to y" by finding an appropriate y. This was done by finding the y that made the direction between x and y as similar as possible to the direction between man and woman. A second constraint was that x and y had to be near enough to each other in the vector space that they were likely to have related meanings. By sampling random x values and finding the most appropriate y, they were able to generate an arbitrary number of analogies for each of the seed word pairs.

Upon generating these analogies, they crowd-sourced reviews from Amazon Turk workers. The two questions they asked were "do the analogies make sense?”, which suggests the embedding is useful, and “do the analogies capture a gender-biased stereotype?”, which is bad. Each analogy was rated by ten workers and the degree of bias was judged by how many considered the statement biased. The results showed that, “overall, 72 out of 150 analogies were rated as gender-appropriate by five or more crowd-workers, and 29 analogies were rated as exhibiting gender stereotypes by five or more [out of ten] crowd-workers.“

Upon demonstrating that bias is present, the authors sought to modify the embeddings so the gender bias was no longer as pronounced but the embeddings were still useful for completing associations. To accomplish this, they first determined which words were definitionally-gendered, by manually annotating a subset and then training a classifier to identify the rest of them. Then they took the words which were not definitionally gendered and sought to reduce or eliminate their component along the gendered direction. This new embedding was used to produce analogies which were also analyzed by crowd workers. The results suggested that the debiased word embeddings generally maintained their ability to solve useful analogies while having significantly-fewer biased analogies.

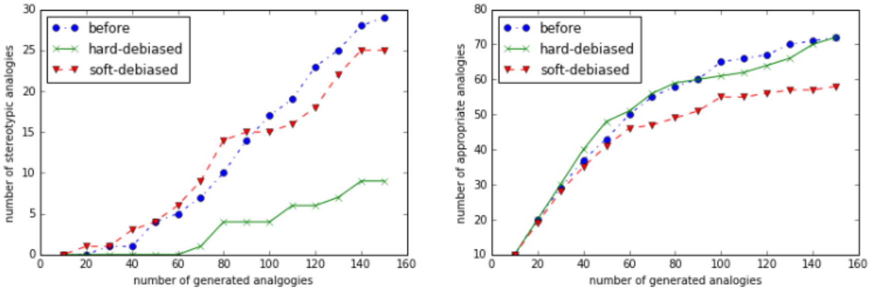
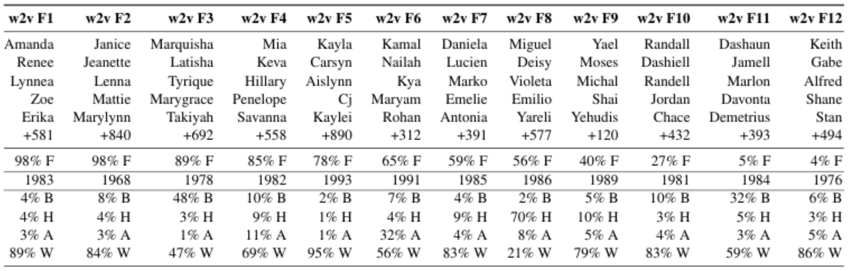


Figure 3: Number of stereotypical (Left) and appropriate (Right) generated before and after debaising. Taken from [1]



w2v F1	**w2v F2**	**w2v F3**	**w2v F4**	**w2v F5**	**w2v F6**	**w2v F7**	**w2v F8**	**w2v F9**	**w2v F10**	**w2v F11**	**w2v F12**
Amanda	Janice	Marquisha	Mia	Kayla	Kamal	Daniela	Miguel	Yael	Randall	Dashaun	Keith
Renee	Jeanette	Latisha	Keva	Carsyn	Nailah	Lucien	Deisy	Moses	Dashiell	Jamell	Gabe
Lynnea	Lenna	Tyrique	Hillary	Aislynn	Kya	Marko	Violeta	Michal	Randell	Marlon	Alfred
Zoe	Mattie	Marygrace	Penelope	Cj	Maryam	Emelie	Emilio	Shai	Jordan	Davonta	Shane
Erika	Marylynn	Takiyah	Savanna	Kaylei	Rohan	Antonia	Yareli	Yehudis	Chace	Demetrius	Stan
+581	+840	+692	+558	+890	+312	+391	+577	+120	+432	+393	+494
98% F	98% F	89% F	85% F	78% F	65% F	59% F	56% F	40% F	27% F	5% F	4% F
1983	1968	1978	1982	1993	1991	1985	1986	1989	1981	1984	1976
4% B	8% B	48% B	10% B	2% B	7% B	4% B	2% B	5% B	10% B	32% B	6% B
4% H	4% H	3% H	9% H	1% H	4% H	9% H	70% H	10% H	3% H	5% H	3% H
3% A	3% A	1% A	11% A	1% A	32% A	4% A	8% A	5% A	4% A	3% A	5% A
89% W	84% W	47% W	69% W	95% W	56% W	83% W	21% W	79% W	83% W	59% W	86% W

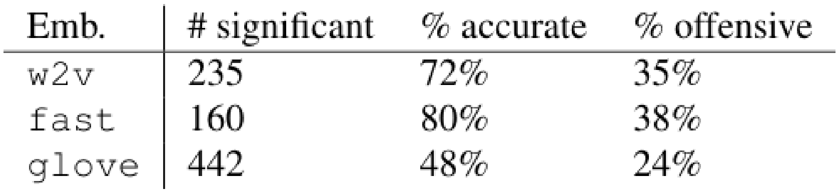
Figure 4: Names clusters obtained from the embeddings. Demographic characteristics included below are computed post-hoc and not used for clustering. Taken from [15]

A limitation of [1] is that it cannot detect other sorts of bias such as racism in embeddings. The work of [15] seeks to extend the analysis to other attributes and also do so without pre-specifying which attributes to evaluate. This work specifically looks at the biases in name embeddings because biased names could have major ramifications on search algorithms and resume classification.

The first discovery they make is realizing that simple K-means clustering exposes groups of names which have internally-similar demographic connotations, as seen in Figure 4. They also find that clustering all words excluding names also reveals semantically-relevant clusters, such as those related to food, style, family and occupation. To analyze the bias in the embeddings, they appeal to a previously-proposed Word Embedding Association Test (WEAT) which describes the level of relation between two clusters of names and two corresponding sets of words. The WEAT measures the mean difference between the names multiplied by the difference between the sets of words. This can be thought of as similar to the strength of the analogy discussed in the Man is to Woman paper.

To better understand the associations, they study each type of word individually. For example, consider all of the words which are in the food cluster. They now want to figure out which set of names each food is most strongly associated with. To do this they simply compute the distance to each of the mean names and associate the food with that name. Now they have partitioned the names into k sets, corresponding to the k different sets of names. The WEAT statistic with these two sets, each containing k-subsets, is computed to determine the degree of association between the names and the foods which are nearest each name.

Now the authors seek to determine which types of words are meaninging-fully associated with the sets of names. To do this they seek to compute the association scores under the null hypothesis, which would be that there is no association between the names and words within that set. Because there is no closed-form solution to this, they must estimate this distribution using Monte Carlo sampling. They determine that an appropriate way to represent the null hypothesis is to shift the embeddings relative to the names a random amount and compute the WEAT statistic with this shifted data. They draw rotation matrices uniformly from the Haar measure, which ensures uniform distribution over rotations about all axes, and multiply all word embeddings by this matrix. Now they compute a p value for the original WEAT which states the fraction of randomly-drawn associations have a larger WEAT value. If the p value exceeds a certain threshold, chosen to account for the desired false discovery rate and the number of multiple comparisons, the association is strong enough that it is unlikely to have been caused by chance.



Emb.	# significant	% accurate	% offensive
w2v	235	72%	35%
fast	160	80%	38%
glove	442	48%	24%

Figure 5: The percent accuracy and the percent of offensive associations for three embeddings. Taken from [15]

After collecting the associations with statistically-significant WEAT scores, they present these k sets of names and k subsets of a type of word to Amazon Mechanical Turk workers. The workers are asked to make the associations which they think are most stereotypical between the names and the words. From this they are able to tell how many of the generated WEATs match the workers response. For all generated associations the crowdworkers agree with, they then ask other workers to rate them in terms of offensiveness. The results of these two experiments, for each of three embeddings, is presented in Figure 5. This suggests that embedded names carry some sort of connotation, which could be related to gender, race, age, or another factor, or a combination thereof. Furthermore, in a large fraction of these cases, these associations perpetrate offensive stereotypes.

6 Incompleteness of Debiasing

To the best of our knowledge, no serious work has sought to refute the presence of bias in embeddings. However, a point of contention is whether debiasing algorithms, often posed to mitigate or eliminate one form of bias, are actually effective. In their 2019 work *Lipstick on a Pig: Debiasing Methods Cover up Systematic Gender Biases in Word Embeddings But do not Remove Them* [6], Gonen and Goldberg compellingly argue that popular gender debiasing algorithms only obfuscate the bias and do not meaningfully remove it. They specifically focus on the approach proposed by [1], described above, and a modification to the training procedure for the GLoVe embedding algorithm, GN-GLoVe [16]. The latter seeks to concentrate all of the gendered information in the last component of the embedding vector so it can be easily removed in applications which should be gender-blind.

Gonen and Goldberg focus solely on gender and contend that even for this relatively simple single-attribute problem, the current definition of bias is incomplete. As proposed by [1], the commonly accepted notion of bias is how strongly words which should not be gendered, such as occupations, have projections onto the primary gender axis of the embedding space. The Man is to Woman paper briefly notes that reducing this projection does not eliminate the

H

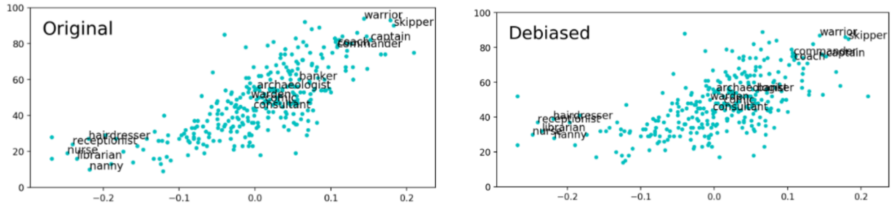


Figure 6: Words still have a high proportion of same-gender-associated neighbors even after supposed debiasing with HARD-DEBIASED. The x axis is the degree of bias in the original embeddings and the y axis is the fraction of female-embedded neighbors, computed from the original embedding. Taken from [6]

clusters between non-gendered words, such as female-stereotyped occupations being embedded closely. While this is largely dismissed as a subtlety in the original work, the authors of [6] seek to show that these clustering effects makes debiasing algorithms relatively ineffective.

In the first experiment, they take the 500 most strongly male- and female-biased words from the original embedding and seek to cluster them before and after the debiasing step is completed. They use K-means with two clusters and compare the word's cluster labels to the original male-female characterization. Unsurprisingly, these original clusters are 99.9% consistent for the word2vec embeddings and 100% consistent for GLoVe, suggesting significant differences between these male and female embeddings. After debiasing word2vec with the HARD-DEBIASED algorithm proposed by [1] and GLoVe with the GN-GLoVe, the clusters were still 92.5% and 85.6% consistent. While this shows that debiasing can help, this result is troubling because it means that significant gendered associations were maintained for many words after they were supposedly de-gendered.

In a similar vein, they show that there is correlation between the original gender projection and the fraction of nearby words which are gendered. This remains even after the de-biasing step, suggesting that female-stereotype words such as nurse and hairdresser remain neighbors after the embedding. Figure 6 demonstrates this property. Finally, they take 5000 of the most gendered words from the embeddings and split them into a training and test set of 1000 and 4000 words, respectively. Then, they train a RBF-kernel SVM on the training data to separate male and female words and use this to predict the gender of the samples in the test set. They achieve 98.25% and 98.65% accuracy on the original versions of word2vec and GLoVe, respectively. However, they still achieve a surprisingly high accuracy, at 88.88% and 96.53% respectively on the supposedly debiased embeddings.

Overall, this work suggests that even though it may be possible to reduce the most visible signs of bias, this is an incomplete solution. They do not state that debiasing with these methods hurts, and in fact it does seem to help somewhat, but the improvements are much more slight than prior work advertises. Similar to the need for many domain-specific definitions of fairness, one should be cautious of an approach which suggests they have an all-encompassing definition or solution to bias in word embeddings.

7 Conclusion

Word embeddings are a powerful and useful tool when seeking to understand written text. However, since they are trained on examples of how language is used in practice, embeddings are almost guaranteed to have biased properties. This bias can perpetuate and amplify stereotypes, which is especially harmful in highly regulated spaces such as hiring and criminal justice. While substantial work has been done on seeking to reduce bias, there are strong arguments that these methods are insufficient. They primarily hide the negative effects and do little to reduce the harmful impacts. Bias in NLP is not caused by bad algorithms, but by biased datasets. Common datasets include wikipedia and google news, common sources of seemingly unbiased and truthful information. What bias in NLP really tells us is that there is bias in society. In fact, NLP is a useful tool for studying societal biases from a social and linguistic standpoint. NLP has helped us understand the nature and characteristics of sexual harassment, the tie between names and racial, ethnic, and gender stereotypes, and given us a more nuanced understanding of linguistices and bias.

However technologists need to be aware of, and take responsibility for, when the technologies that they build perpetuate inequalities and harm people. Lack of expertise in a particular domain does not release technologists from responsibility. It is the responsibility of technologists to actively seek collaboration from domain experts and stakeholders. In order to prevent future harm due to biased algorithms, our field must set a standard of ethical guidelines with accountability for companies and developers. Algorithms should always uphold the human values of the domain for which they are built for. If algorithms cannot uphold these values, then the particular decision-making space should not be automated. There needs to be incentives for iterative improvement of algorithms used in critical decision-making spaces, with stakeholders given the information used to make this particular decision about them as well as what factors led to the outcome. All algorithms developed with the purpose of working in the decision-making landscape, especially in regulated areas, should release a social impact report, with detailed analysis of input data, the definition of fairness that was defined by domain experts, and a report of the effectiveness of the algorithm in reaching this definition of fairness.

References

[1] Tolga Bolukbasi et al. *Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings.* Ed. by D. D. Lee et al.

2016. URL: http://papers.nips.cc/paper/6228-man-is-to-computer-programmer-as-woman-is-to-homemaker-debiasing-word-embeddings.pdf.

[2] Brownlee. *What Are Word Embeddings for Text?* 2017. URL: https://machinelearningmastery.com/what-are-word-embeddings/.

[3] Robyn Caplan et al. *2018*. URL: https://datasociety.net/library/algorithmic-accountability-a-primer/.

[4] Avnaim-Pesso Danziger Levav. “Extraneous factors in judicial decisions”. In: *PNAS* (2011).

[5] Jeffrey Dastin. *Amazon scraps secret AI recruiting tool that showed bias against women*. 2018. URL: https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G.

[6] Hila Gonen and Yoav Goldberg. *Lipstick on a Pig: Debiasing Methods Cover up Systematic Gender Biases in Word Embeddings But do not Remove Them*. Mar. 2019.

[7] Sarah Harrison. *Five Years of Tech Diversity Reports - and Little Progress*. 2019. URL: https://www.wired.com/story/five-years-tech-diversity-reports-little-progress/.

[8] Intellica. *Comparison of different Word Embeddings on Text Similarity — A use case in NLP*. 2019. URL: https://medium.com/@Intellica.AI/comparison-of-different-word-embeddings-on-text-similarity-a-use-case-in-nlp-e83e08469c1c.

[9] Jon Kleinberg et al. “Discrimination in the Age of Algorithms”. In: *Journal of Legal Analysis* 10 (2018), pp. 113–174. DOI: https://doi.org/10.1093/jla/laz001.

[10] Derek Leben. “Normative Principles for Evaluating Fairness in Machine Learning”. In: *2020 AAAI/ACM Conference on AI, Ethics, and Society (AIES ’20)*. AAI/ACM. 2020.

[11] Yingchi Liu et al. *Uncover Sexual Harassment Patterns from Personal Stories by Joint Key Element Extraction and Categorization*. Nov. 2019.

[12] Lynn. *Get Busy with Word Embeddings – An Introduction*. 2018. URL: https://www.shanelynn.ie/get-busy-with-word-embeddings-introduction/.

[13] Tomas Mikolov et al. *Efficient Estimation of Word Representations in Vector Space*. 2013. URL: http://arxiv.org/abs/1301.3781.

[14] Martin Mitchell. “Gender Bias in Student Evaluations”. In: *American Political Science Association* (2018). DOI: doi:10.1017/S104909651800001X.

[15] Nathaniel Swinger et al. “What are the biases in my word embedding?” In: *CoRR* abs/1812.08769 (2018). arXiv: 1812.08769. URL: http://arxiv.org/abs/1812.08769.

[16] Jieyu Zhao et al. “Learning gender-neutral word embeddings”. In: *arXiv preprint arXiv:1809.01496* (2018).