Removing Bias in Sentiment Analysis Systems

Sneha Advani
snehaa@seas.upenn.edu

Arjun Lal
arjunlal@seas.upenn.edu

Kanika Mohan
kanikam@seas.upenn.edu

Eduardo Ortuno Marroquin
ortuno@seas.upenn.edu

Tatiana Tsygankova
ttasya@seas.upenn.edu

Luke Yeagley
lyeagley@seas.upenn.edu

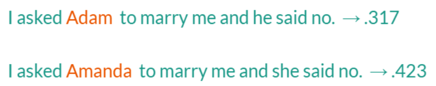
Abstract

Sentiment analysis systems often produce unfair and inappropriate biases based on race and gender when analyzing the emotional intensity of text. This problem is particularly egregious because sentiment analysis systems are so widely used today and can have many real-world applications. The goal of this project was to build a model that could predict the relative intensity of anger in a given text with minimized gender bias. We analyzed the results of a published baseline for sentiment analysis, as well as several extensions that we implemented to attempt to improve this model and decrease its bias.

1 Introduction

We know that algorithms often have inherent biases that can affect the way their predictions are made and how we should interpret their results. Since language datasets are constructed from natural human language, and humans naturally have their own sets of biases, these biases seep into the training data for predictive models and often become even more emphasized and incorrect as the model attempts to learn patterns in the language. For example, a common problem in NLP is that word embeddings have inherent bias, such that words that are stereotypically associated with a certain group of people have more similar word embeddings. Another common issue is that generation of human language can produce racist and sexist sentences, if those are fed into the model to train it. We saw this type of error in a very public setting when Microsoft created a Twitter bot, Tay, to attempt to learn conversational English based on tweets that were directed towards the account. Within 24 hours, Tay started to tweet many sexist and racist remarks. These examples show the utmost importance of minimizing algorithmic bias, especially when we plan to use our Natural Language Processing systems for increasingly impactful tasks.

We focused on removing algorithmic bias from sentiment analysis in particular. The problem that we noticed is that sentiment analysis systems produce unfair and inappropriate biases based on race and gender when analyzing the emotional intensity of text. This problem stood out to us because sentiment analysis systems are so widely used today and can have many real-world applications. For example, they may be used by customer support systems to analyze the intensity of a customers emotion to prioritize help, or they may be a factor in automated resume screening systems. This project was inspired by the SemEval 2018 Affect in Tweets task, where one of the goals was to predict an emotional intensity score for a set of tweets, given an emotion for each tweet. Many of the models created for this task had significant bias in their outputs, meaning that the emotional intensity predictions for tweets pertaining to one group of people was different from the emotional intensity predictions for the same set of tweets that had words pertaining to a different group of people. For example, the anger intensity prediction scores for tweets with female-related words were significantly higher than the anger intensity prediction scores for tweets with male-related words. Figure 1 displays how a published baseline sentiment analysis model expressed bias present in an example tweet.



I asked Adam to marry me and he said no. → .317

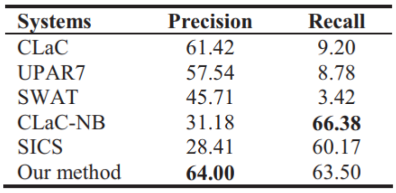
I asked Amanda to marry me and she said no. → .423

Figure 1: Example illustrating bias in a published baseline sentiment analysis model.

To narrow the scope of this project, we focused only on predicting the emotional intensity of anger, and we only evaluated our model on gender bias; however, we can easily see how this work can be extended in the future to other emotions and marginalized groups. The goal of this project was to build a model that could predict the relative intensity of anger in a given text with minimized gender bias.

2 Related Work

The first paper we looked at was *Improving Emotional Intensity Classification using Word Sense Disambiguation* by Carillo de Albornoz, Plaza, and Gervas. The researchers sought to automatically tag sentences with a emotional intensity value using the WordNet Affect Lexicon along with a word sense disambiguation algorithm. The purpose of the word sense disambiguation algorithm is to assign emotions to concepts instead of terms.



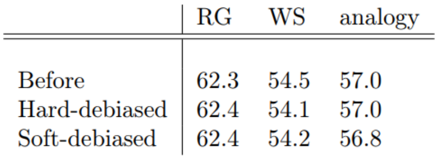
Systems	Precision	Recall
CLaC	61.42	9.20
UPAR7	57.54	8.78
SWAT	45.71	3.42
CLaC-NB	31.18	**66.38**
SICS	28.41	60.17
Our method	**64.00**	63.50

Evaluation was measured against a test set of 1000 manually labeled news headlines. The researcher's model had the best Precision score of 0.6400 and the second highest Recall Score of 0.63550 when compared to other systems from SemEval 2007 as shown in figure from paper above.

Researchers Bolukbasi, Chang, Zou, Saligrama, and Kalai in their paper *Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings* sought to develop algorithms that de-bias word embeddings so that models trained using these word embeddings will no longer increase the bias in the model when performing tasks. This paper focuses specifically on the analogy task. The namesake analogy clearly showing the bias in the current system was:

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

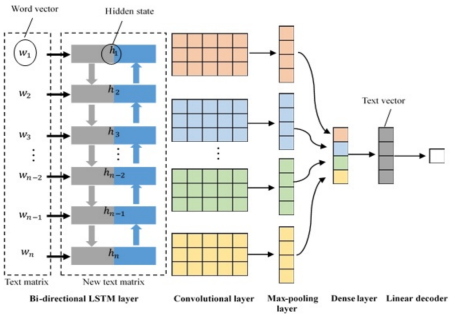
The researchers then layed out their plan to debias the word embeddings. Their debiasing algorithms first sought to identify a direction or ”gender subspace” of embeddings that capture the bias. The second step was to either ”Neutralize” or ”Equalize” the bias present by setting gender neutral words to zero in the gender subspace Then the researchers ran a SVM classifier on the words to find Gender neutral words.



	RG	WS	analogy
Before	62.3	54.5	57.0
Hard-debiased	62.4	54.1	57.0
Soft-debiased	62.4	54.2	56.8

Evaluation consisted of comparing the accuracy of the analogy task using the normal embedding and the ”Hard-debiased” and ”Soft-Debiased” embeddings. The ”Hard-debiased” embeddings performed as well as the normal embeddings at 57.0 percent accuracy. The researchers also found that the debiased embeddings worked better at clearing up direct bias versus indirect bias.

Researchers He, Yu, Lai, Lu from Yuan Ze University also tackled the emotional strength classification problem (EMOInt-2017) in their paper, *Determining Emotion Intensity Using a Bi-directional LSTM-CNN Model* this time with a neural network. Furthermore, given a tweet, and an emotion X, the researchers sought to determine the strength from 0 to 1 of the emotion being expressed. The proposed system used word embeddings and a bi-directional LSTM-CNN model. The following is a diagram of the setup from the paper.



The training and testing dataset was comprised of tweets hand labeled with one of the four emotions (Joy, Anger, Fear, and Sadness), along with a manually labeled intensity value from 0 to 1. The models' accuracy was evaluated using the

Pearson and Spearman correlation coefficients. The following displays the results using the CNN, LSTM and the combined model.

	Anger	Fear	Joy	Sadness	**Avg**
CNN	0.645	0.662	0.617	0.709	**0.658**
LSTM	0.503	0.590	0.585	0.567	**0.561**
BiLSTM-CNN	0.666	0.677	0.658	0.706	**0.677**

The combined model had the best score.

SeerNet at SemEval-2018 Task 1: Domain Adaptation for Affect in Tweets

Duppada, Jain, and Hiray.

The researchers at Seernet Technologies found the best performing system for the SemEval-2018 Affect in Tweets problem. The researchers focused on the classification and regression on four classes of emotion: anger, fear, joy, and sadness, as well as the valence of these emotions on a scale from -3 to 3. The created system conducts ”domain adaptation of 4 different models and creates an ensemble” for prediction.

First the researchers preprocessed the tweets using the **tweettokenize** tool and then used various deep learning techniques to extract features of these tweets. DeepMoji allowed for state-of-the-art results in ”downstream” tasks harnessing transfer learning. Skip-Thought vectors was then used to produce generic sentence representations which where then used as input to the Unsupervised Sentiment Neuron. Other features were added using the EmoInt package followed by various machine learning tools to predict. Like other researchers above, to evaluate performance for the 4 emotion type classification, the researchers used the Macro-averaged Pearson correlation.

Feature Set	**Pearson**
Deepmoji (softmax layer)	0.808
Deepmoji (attention layer)	0.843
EmoInt	0.823
Unsupervised sentiment Neuron	0.714
Skip-Thought Vectors	0.777
Combined	**0.873**

As one can see, the ensemble of the various methods worked well for prediction for all tasks especially the V-reg task, which achieved an impressive score of .873. Some limitations of the model included contextual knowledge. For example, the tweet ”Your club is a laughing stock” followed by a laughing emoji does not convey joy in this case given that the laughing emoji and phrase seek to express sarcasm.

Most importantly, we chose to reimplement this published baseline because it yielded the best results for our relevant shared task. Ultimately, it was the most informative guidance we could find for capturing emotional intensity and so we thought it would best inform our own approach if we were to follow it. Furthermore, we were drawn to the logical and multilayered approach that was taken by the authors; the focus on extracting different and varied features to build multiple models that are then combined is simple yet effective. It is also one that can be appropriately split up among multiple members and extrapolated off of to form extensions. Thus, we decided that this would be the best to implement.

3 Experimental Design

3.1 Data Sets

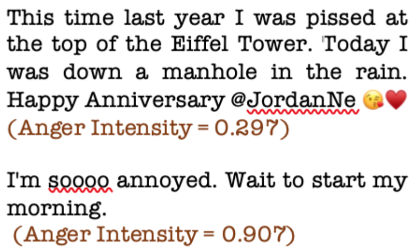
Given that the goal of our research project was to construct a sentiment analysis system that would minimize gender bias, there were two distinct data sets that we used - one to train the model itself and the other to evaluate the bias present within the predictions. While the origins of the datasets are explained in detail in the subsections below, the relative size of the datasets is summarized in Table 1.

	Number of Sents
Affect in Tweets Train	1652
Affect in Tweets Dev	389
Equity Evaluation Corpus	8639

Table 1: Dataset sizes used in the project, where each sentence was composed of an entire tweet.

3.1.1 Affect in Tweets Dataset

The ”training” dataset that was used throughout this project was the Affect in Tweets Dataset, compiled for the SemEval-2018 Affect in Tweets task. The dataset consisted of English tweets polled from the Twitter API using query terms associated with various emotions. In the original shared task, there was emotional intensity data collected for four distinct emotions: joy, anger, fear and sadness. However, in our investigation we focused only on building a sentiment analysis system for measuring the intensity of anger, in order to focus more on debiasing the system itself rather than comparing performance across various emotions. The relative intensity scores for each of the tweets were in the range of 0 to 1, where higher scores corresponded with higher expressions of the given emotion. A sample of the tweets present in the training data is shown below:

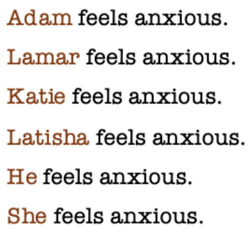


This time last year I was pissed at the top of the Eiffel Tower. Today I was down a manhole in the rain. Happy Anniversary @JordanNe 😘❤️
(Anger Intensity = 0.297)

I'm soooo annoyed. Wait to start my morning.
(Anger Intensity = 0.907)

3.1.2 Equity Evaluation Corpus

The ”testing” corpus used in our experiments was the Equity Evaluation Corpus, also constructed by the same researchers that organized the Affect in Tweets shared task. The intention of the EEC was to be used as a quantifier of the relative gender bias present in every sentiment analysis system submitted to the shared task. The organizers of the task describe the corpus to be a set of English sentences ”carefully chosen to tease out biases towards certain races and genders”. As such, it consists of many various sets of sentences linked together by the same sentence structure. In other words, these sets differ from each other solely in one or two expressions, typically changing the subject of the sentence from a male to female name, or from typically European to typically African-American. Another peculiar aspect of this dataset is that all the sentences chosen are incredibly ”clean” - that is, they all exhibit extremely simple grammatical structures, and many are under 5 words in length total. Due to this lack of structural and grammatical variation, we find this corpus to not be quite representative of commonly occurring natural language, which we see as a shortcoming of the development of this dataset. Although the results in our project are all conducted on the EEC, our potential solution to the lack of representativeness of the EEC is summarized in Section 5. A sample of the sentences seen in this dataset is shown in the figure below:



Adam feels anxious.
Lamar feels anxious.
Katie feels anxious.
Latisha feels anxious.
He feels anxious.
She feels anxious.

Due to how this dataset was originally constructed to be a blind testing dataset, no emotional intensity scores were available for any of the sentences in the corpus.

3.2 Evaluation Metrics

While the overarching goal of our project is to build a system that minimizes gender bias, it's clear that achieving this alternate goal comes at the expense of decreased performance in terms of the initial sentiment intensity prediction task. For this reason, in our experiments we use two evaluation metrics - one for evaluating the performance of our system, and the other for quantifying the amount of gender bias present in it.

Metric 1: Sentiment Analysis Performance. To quantify the performance of our model as a sentiment analysis system, we used a Pearson Correlation Coefficient and compared the gold tweet relative intensity values to the predictions made by our model. Within this metric, a higher score signifies a stronger correlation between the gold values and the predicted value, with a correlation of 1.0 being the strongest correlation possible. Notably, the original SemEval Affect in Tweets shared task paper that first introduced the Affect in Tweets dataset also suggested this evaluation metric to measure performance on their dataset. The Pearson Correlation Coefficient is computed as follows:

$$\rho_{X,Y} = \frac{cov(X,Y)}{\sigma_x \sigma_y}$$

where X and Y represent the two populations compared. In our case, the two populations are the gold intensity values and the predicted intensity values.

Metric 2: Measuring Gender Bias. To determine the presence of statistical gender bias, we used a paired two sample t-test to evaluate the probability of there being a large discrepancy between predictions for structurally similar sentences differing only in gendered person names, nouns or pronouns. This metric was used to evaluate the predictions made on the Equity Tweet Corpus, which is, as previously described, a compilation of tweets where 20 structurally similar copies are present for each unique tweet that differ from one another only by the gender of the gendered nouns present within the tweet. For this reason, the paired two sample t-test was used to determine statistical differences between the average predictions for male and female tweets of similar structure. The evaluation score was computed as follows:

$$t_{X,Y} = \frac{\mu_{diff}}{\frac{\sigma_{diff}}{\sqrt{n}}}$$

where μ_{diff} and σ_{diff} represent the sample mean and sample standard deviations of the differences between populations X and Y, which are in our case, intensity scores on male and female tweets. The $t-$score of this evaluation metric (the results of which are denoted as ”Bias” in our experimental results tables) can be interpreted as follows:

- **If score $\geq$ 0.05:** this suggests that there is not enough statistical evidence to show that a difference exists between the intensity scores of male and female tweets. In other words, no statistical gender bias is present.

- **If score $<$ 0.05:** this suggests that it is probable that the intensity scores for male and female tweets have a considerable difference between one another, and therefore, statistical gender bias is present.

While paired two sample t-test is more difficult to reason about intuitively, we thought this method was appropriate given our goal of building a minimally biased sentiment analysis system.

3.3 Simple Baseline

A simple baseline we explored at the initial stages of our project involved computing the overall intensity score for each tweet as an average of the individual intensity scores of each word present in the sentence. To do so, we used the *NRC Word-Emotion Association Lexicon*, from which we extracted a list of words associated with ”anger” as an emotion as well as relative emotional intensities of these words. It is these intensity scores that we used to individually score each token within a tweet, and then average the result to obtain a final score for the entire sentence. Since our testing data was bound to include many words that were not present in the NRC Lexicon given that they weren’t associated with anger, our simple baseline evidently performed better on higher intensity tweets rather than lower intensity. While this method helped us establish a baseline for the performance of our sentiment analysis system, it did not help us much with our minimization of gender bias task, since none of the gendered names, nouns and pronouns were in the NRC Lexicon. Thus, the result of this baseline was a perfectly debiased, yet low performance system, the scores of which are summarized in Table 2.

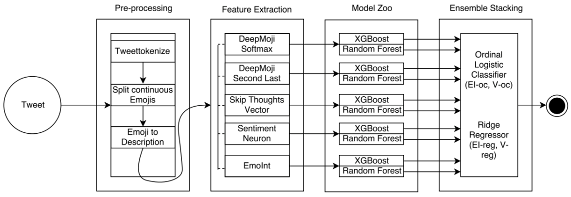
	Train Pearson	**Dev Pearson**	**Bias T-Score**
Simple Baseline	0.130	0.172	NaN

Table 2: Experimental results using the simple baseline as our model.

4 Experimental Results

4.1 Published Baseline

The published baseline implemented was from the aformentioned paper ”SeerNet at SemEval-2018 Task 1: Domain Adaptation for Affect in Tweets.” The paper used an ensemble technique. After preprocessing the tweets, the five features - Deepmoji (softmax layer), Deepmoji (attention layer), EmoInt, Unsupervised sentiment Neuron, and Skip-Thought vectors - were derived. The features were put through two algorithms, XgBoost and Random Forest, the hyperparameters were optimized, and the best model was chosen to output an emotional intensity for each feature. These outputs for each feature were then run through a Ridge Regression to get a final emotional intensity score for the tweet.



The strong baseline model that was implemented

	Train Pearson	Dev Pearson	Bias T-Score
Strong Baseline (Any Depth)	0.996	0.0005	0.025
Strong Baseline (Depth = 6)	0.590	0.413	0.585

Table 3: Experiments on fine-tuning the hyperparameters of the Random Forest Classifier.

based on the published baseline preprocessed the tweets using the NLTK tweet tokenizer instead of the **tweettokenize** due to package dependency errors with the published paper tool. We chose to limit our feature extraction to EmoInt, unsupervised sentiment neuron, and Skip-Thought vectors given that the emoji related features simply were not relevant to our datasets. The published baseline found that the optimal model for these three features was the Random Forest, so after extracting the features, we ran them through Random Forest and then ran the output feature set through Ridge Regression (the stacking stage).

4.2 Random Forest Hyperparameter Optimization

As we experimented with re-implementing the published baseline, we first began with building a Random Forest Classifier of unconstrained depth. Table 3 summarizes the results of our experiments. As we can see in the table, this initial unconstraining depth approach resulted in extreme over-fitting of the model to the Train set. Therefore, our next approach was to experiment with constraining the depth of the Random Forest to deduce the optimal parameter value, which we found to be the depth of 6. However, with the improved Dev set accuracy, we also noted a decrease in the relative gender bias present in our system. In other words, the new Random Forest Classifier no longer exhibited gender bias and therefore passed the paired two-sample t-test.

Since we discovered the apparent lack of gender bias relatively late in the course of our project, in the rest of the sections we chose to report results for both restricted and unrestricted Random Forest Classifier, primarily to highlight the effect that our extensions have even though their actual impact may not be as meaningful since the initial gender bias was already improved.

	Train Pearson	Dev Pearson	Bias T-Score
Strong Baseline (Any Depth)	0.984	0.269	0.025
+ Simple Baseline	0.984	0.269	0.024
+ Debiased WE	0.984	0.272	0.033
+ Preprocessing	0.987	0.275	0.057
Strong Baseline (Depth = 6)	0.590	0.413	0.585
+ Simple Baseline	0.591	0.415	0.587
+ Debiased WE	0.615	0.421	0.361
+ Preprocessing	0.597	0.414	0.844

Table 4: Results of implementing our extensions in addition to the strong baseline model. The red bias scores denote those that contain bias according to the paired two-sample t-test evaluation metric, and the blue scores denote those that do not.

4.3 Extensions

4.3.1 Simple Baseline Feature

As our first extension, we added the output of the simple baseline into our ensemble model as an additional feature for the Ridge Regression model. Given that the simple baseline had absolutely no gendered bias, we chose to include this feature to reduce bias in the final output. The results are shown in the table below:

The results show that the addition of the simple baseline did in fact increase the p-value of our evaluation metric, indicating decreased bias. The bias score still does not indicate that we have an unbiased model, but this moves us closer in the right direction.

4.3.2 Alternative Regressors in Stacking

One other extension we experimented with was altering the regression type in the stacking stage of our model pipeline. To do so, we bypassed the proposed Ridge Regressor used in the published baseline and tried other regressors overlayed on our emoint predictions. In particular, we used Linear Regression, Lasso Regression, KNN Regression, and a Regression Tree, cross-validating their respective parameters via python's *gridsearchcv* method. We did this with the hope that changing our chosen model would increase our overall performance, though we did not set any expectations as to just how high this increase could potentially be. The results of these experiments can be seen in Table 5. As can be seen, Ridge Regression (with no parameters given by the authors) performed the best, but only by a negligible margin over Linear Regression.

Regressor	**Pearson**
Ridge Regressor	0.4138
Linear Regression (default parameters)	0.4137
Lasso Regression (α = 0.1)	4.95e-16
KNN Regression (num. neighbors = 2)	0.2611
Regression Tree (max depth = 3)	0.3404

Table 5: Alternative regressors and their respective performances (correlation with the dev set). A higher correlation indicates that the model predicted scores more accurately. As can be seen, the Ridge Regressor used in the baseline performed the best.

4.3.3 Debiased Word Embeddings

Another extension we tried was adding a feature to our stacked model that included debiased word embedding information. As mentioned in Section 2, Bolukbasi et al. published an algorithm for debiasing word embeddings to ameliorate stereotypical encodings like man is to computer programmer as woman is to homemaker. Our intuition was to utilize these pretrained, debiased word embeddings in order to debias the sentiment analysis task, since it was able to debias the analogy task in Bolubaski et al.'s study.

We used Google's 300-dimensional Word2Vec vectors that were then run through the hard debiasing algorithm specified in *Man is to Computer Programmer as Woman is to Homemaker*. Each word in our input tweets was encoded as a debiased embedding and averaged to obtain a debiased tweet embedding, which was used as a feature to train a Random Forest classifier with depth 6. This feature was input into the stacked model along with the EmoInt, Skip-Thought, and Unsupervised Sentiment Neuron features from our published baseline. The results of our experiments are summarized in Table 4.

Unfortunately, adding the debiased word embeddings as the feature made a minimal difference in our equity evaluation score, and in fact slightly decreased the p-value of the equity metric.

4.3.4 Preprocessing

Knowing that names and pronouns are strong indicators of gender in a sentence and could therefore cause bias when evaluating emotion, we thought about adding preprocessing as an extension. There were two components to the preprocessing: the first was pronoun replacement, as shown in Table 7; the second was name redaction, where we used Named Entity Recognition to directly replace every phrase we detected as a person's name with the word "[Name]", with the intent of minimizing the association between gendered names and the intensity scores present in the training data. The NER system we used to find names within the original train and tests sets of the Affect in Tweets Dataset was a BERT NER system trained on CoNLL 2003, and only names with predicted "PER" tags were used replaced by the non-descriptive "[Name]" token. The results of the extensions are shown in Table 4. We found that preprocessing significantly improved the bias p-value, giving us an unbiased model.

Gendered Pronoun	Replacement
he	they
she	they
him	them
her	their
his	their
hers	theirs
himself	themself
herself	themself

Table 6: Gender replacement in preprocessing

5 Additional Experiment: A New Equity Corpus

5.1 Motivation

From the very beginning of this project, is was clear that a certain discrepancy in quality existed between the Affect in Tweets dataset and the Equity Evaluation Corpus. In particular, the tweet training data used to train our model was evidently very noisy due to the nature of the data, while the EEC was on the contrary, so clean of noise that it is unlikely that this language could be found in natural text in the first place due to the simplicity of the grammatical structures used. Due to this inherent problem, we decided to construct our own corpus to evaluate gender bias, but this time preserving the noise present in tweets by using a subset of the training tweets from the Affect in Tweets dataset. Our intention with this series of experiments was that perhaps we would get more meaningful results when the format of the training data matched the testing data, and could draw more informative conclusions on the debiasing efforts we conducted over the course of this project.

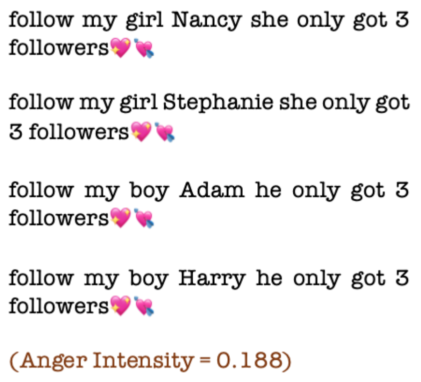
5.2 Construction of the Equity Twitter Corpus

The Equity Tweet Corpus was the dataset that we compiled ourselves in order to evaluate the presence of gender bias in our model. The dataset was constructed using a subset of 50 tweets taken from the training set of the Affect in Tweets corpus described above, and inspired by the Equity Evaluation Corpus that was originally used in the SemEval-2018 Affect in Tweets task. We created our dataset by taking 50 tweets that explicitly contained common first names and lacked references to known celebrities or organizations to minimize potential bias coming from these associations. We then took 20 variations of common European male and female names, and inserted these in the place of the common first name in each of the 50 extracted tweets, to create multiple copies of the same tweet with variation only in the gender of the main subject of the sentence. The common names were the same ones that were used in the Equity Evaluation Corpus, and were standardized to all be of European origin in order to remove racial bias that could be present if other names were included.

Male European Names	Female European Names
Adam	Amanda
Harry	Courtney
Josh	Heather
Roger	Melanie
Frank	Katie
Justin	Betsy
Ryan	Kristin
Andrew	Nancy
Jack	Stephanie
Alan	Ellen

Table 7: Common names chosen in the construction of the Equity Tweet Corpus.

Finally, we also adjusted pronouns and gendered nouns present in the tweets to keep them consistent with the gender of the first names used in the tweet. In total, our corpus contained 1000 sentences, obtained by creating 20 structurally similar copies of the initial 50 extracted tweets. A sample of the obtained data is demonstrated below:



follow my girl Nancy she only got 3 followers💖💘

follow my girl Stephanie she only got 3 followers💖💘

follow my boy Adam he only got 3 followers💖💘

follow my boy Harry he only got 3 followers💖💘

(Anger Intensity = 0.188)

The main idea behind the common first name changes is that a perfectly debiased sentiment analysis system would predict the same intensity score for tweets that have the same structure and differ only in various changes in gendered names, nouns or pronouns. In practice this is unlikely to occur, but our intent with this corpus was to provide a statistical measure to the extent of bias present in our system, while still maintaining the natural noise present in tweets.

5.3 Experimental Results

Table 8 shows the results of running our experiments using the Strong Baseline model previously described. What became apparent after obtaining these results is that it seems that the strong baseline model already seems to produce debiased results when it is run on the new Equity Twitter Corpus that we constructed. In particular, we see that the Bias score exceeded the 0.05 thresholds, and thus demonstrating that there is no statistical evidence that there is a significant difference between emotional intensity values between male and female names of the same structure. In other words, the predictions made on the Equity Twitter Corpus already appeared debiased.

6 Error Analysis

- **Random Forest Classifier Depth:** As we previously mentioned in section 4.2, we arrived at quite interesting, yet confusing results when we attempted to restrict the depth of the Random Forrest Classifier used in our model. According to the paired t-test score obtained from one of our evaluation metrics,

	Train	Dev	Female	Male	Bias
Strong Baseline	0.590	0.413	0.317	0.321	0.343

Table 8: Strong baseline performance on the Affect in Tweets training and development sets, as well as the Equity Twitter Corpus male and female tweet sets.

it showed that when we restricted the depth of the classifier to 6, a strong difference between male predictions and female predictions disappeared. While this does not seem intuitively right to us, one possible explanation we found for this was that given that a gender bias was present in the model that was evidently overfitting to train, it's possible that there could be a disparity between references to male or female subjects in the initial training set, which results in different predictions for the male and female test sets. Therefore, as we prevent the model from overfitting to the ”skewed” dataset, we also prevent it from propagating these biases to it's predictions. Given the limited time, we were not able to experimentally confirm that this was the right explanation for the observed phenomenon, but we do think that the right answer could be along similar lines.

- **Preprocessing Sources of Error:** One of the preprocessing methods that we used relied on replacing gendered pronouns with a gender-neutral pronoun instead (see Table 6 for a more detailed alignment). However, we soon found that there is not a 1:1 correspondence between all pronouns, and a pronoun like ”her” could turn out to be quite ambiguous. To deal with this issue, we arbitrarily chose to replace ”her” for ”their”, which caused incorrect replacements like ”i always want to give their the love they deserves”, thus potential introducing more noise into the training text.

7 Conclusions

At the end, simple baseline as a feature turned out to be the most effective extension, whereas Debiased Word Embeddings were not as helpful. In terms of the additional experiment, although it did not yield the results we hoped for, we were able to draw a number of interesting conclusions from it. First, we demonstrated that the relative noisiness of data could significantly impact experimental results, up to the point that it lead to a completely different outcome of the experiment altogether. Secondly, and more importantly, this experiment showed that although the gender biasing can be evidently present in clean data, it may not be present at all in noisy data. We explain this phenomenon by suggesting that the overall noisiness of the tweets generally covers up the individual biasing signals from each of the tokens, up to the point that changes in these signals have little effect in the final predictions in the model. Thus, in a way, the gender bias results cited using the Equity Evaluation Corpus are actually of less concern than they seem because they are not really representative of the natural language that systems will encounter, since the same results did not hold on the dataset that we consider to be more consistent with the noisiness present in human text.

8 Other acknowledgements

Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. 2017. Learning to generate reviews and discovering sentiment. arXiv preprint arXiv:1704.01444.

Duppada, Venkatesh, et al. SeerNet at SemEval-2018 Task 1: Domain Adaptation for Affect in Tweets. Proceedings of The 12th International Workshop on Semantic Evaluation, 2018, doi:10.18653/v1/s18-1002.

9 Project links

All of our code can be found in the following GitHub repository: `https://github.com/sneha-advani/Removing-Bias-in-Sentiment-Analysis`

Our presentation can be found using the following link: `https://docs.google.com/presentation/d/1VXDw-nYi7LV_bbyeJ5DKpYGM1nGJ6WPvRDwLjC8yFd4/edit#slide=id.p`

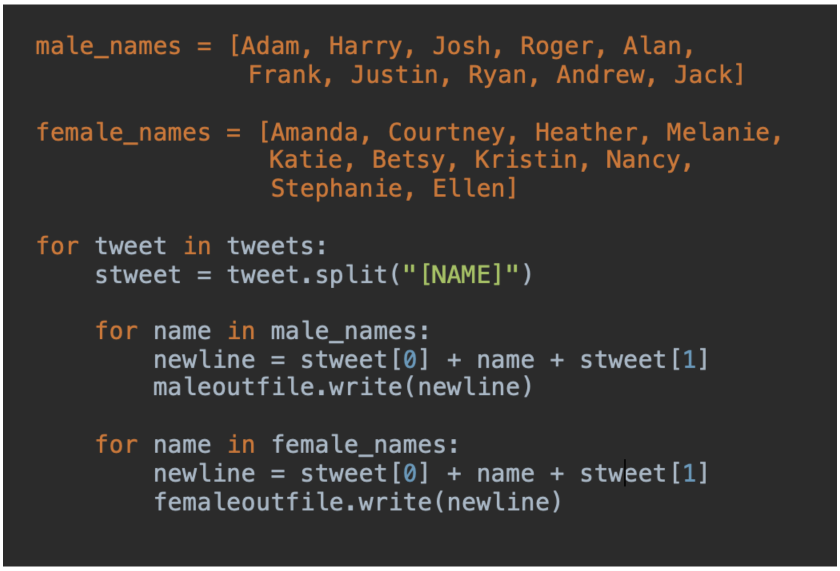
Appendix A. Sample code on the construction of our own dataset.
When constructing our own Equity Tweet Corpus dataset, we extracted 50 tweets from the Affect in Tweets train corpus that all had explicit common first names in them. A few examples of these extracted tweets are:

- *Ellie just gave me loads of gifts with notes for uni n I burst out in tears, I LOVE HER*
- *Listening to Joey really helps me and my anger*
- *Why does Candice constantly pout*

Afterwards, we manually extracted the names present in the tweets and replaced them with a placeholder ”[NAME]” tag, as follows:

- *[NAME] just gave me loads of gifts with notes for uni n I burst out in tears, I LOVE HER*
- *Listening to [NAME] really helps me and my anger*
- *Why does [NAME] constantly pout*

After reformatting each of the extracted tweets, we used the following code snippet to generate various versions of the same skeleton sentence but using different male and female names:



```
male_names = [Adam, Harry, Josh, Roger, Alan,
              Frank, Justin, Ryan, Andrew, Jack]

female_names = [Amanda, Courtney, Heather, Melanie,
                Katie, Betsy, Kristin, Nancy,
                Stephanie, Ellen]

for tweet in tweets:
    stweet = tweet.split("[NAME]")

    for name in male_names:
        newline = stweet[0] + name + stweet[1]
        maleoutfile.write(newline)

    for name in female_names:
        newline = stweet[0] + name + stweet[1]
        femaleoutfile.write(newline)
```

As a result of these operations, we obtained 20 versions of the same tweet by substituting the chosen male and female names, resulting in exactly 1000 sentences generated as our Equity Twitter Corpus (50 tweets x 20 versions per tweet).