Exploring the Relationship Between Intrinsic and Extrinsic Bias in Spanish Word Embeddings

Ricardo Muñoz Sánchez

Master of Science
Artificial Intelligence
School of Informatics
University of Edinburgh
2020

Abstract

Word embeddings are at the heart of the current state of the art in natural language processing. Sadly, they tend to mirror and increase the biases that humans already have. In recent years, there has been research done on how to measure and modify the intrinsic bias in embeddings, however not much of this research has been done in Spanish and none of it has been made on how altering intrinsic bias reflects on extrinsic bias and fairness on downstream tasks.

In this work, we study the relationship between XWEAT, an intrinsic bias metric, and equality of opportunity, an extrinsic bias metric. We use the FastText and word2vec embedding algorithms to be able to compare results and our downstream task is hate speech detection on the HatEval dataset.

We reach the conclusion that the relationship between the two metrics is unreliable, which should call to attention how much we trust them. We also point out the need to develop intrinsic bias metrics that take into account the cultural and linguistic properties of languages other than English, instead of having to rely on translated versions.

Acknowledgements

There are so many people that I would like to thank for their love and support, during the good times, but especially during the hard times.

I would like to start by thanking my supervisors Adam and Seraphina for all their kindness and patience. Their guidance and constant encouragement have helped me to keep going, especially when I was doubting myself and my capabilities.

I also want to thank my teammates, Mugdha and Rebecca. The meetings and conversations we had were constantly reassuring and helped me not feel out of place. I get the impression that we helped keep each other afloat and that is something that I really appreciate.

Another two persons that I would like to thank are David and Gaby, two of my undergraduate teachers. I really value their support and advice throughout my undergraduate degree and the period leading up to me entering this degree. Without them, I would not be here today studying this masters program.

I'm really thankful of my family for their constant support in this new adventure so far from home. They have helped me throughout and have shown me that home need not be a physical space.

Finally, I want to thank the rest of my friends. The old ones, the new ones, the ones I never got a chance to get to know better. Even the ones I lost along the way. This whole academic year has been a roller coaster of emotions and their love and support truly helped get me through even the darkest of days. As Tim McIlrath sang, "but we've had some times, I wouldn't trade for the world."

Table of Contents

Chapter 1

Introduction

There has recently been a meteoric rise in machine learning systems being deployed in real world applications. At the heart of this expansion lie neural networks and big data. However, we do not really know how these systems function. This raises alarming concerns about fairness and discrimination, especially as these systems tend to pick up and amplify biases existing in the data. In natural language processing (NLP), these systems are used in tasks that range from translation to recognizing emotions and categorizing texts.

These have been used on their own, such as Google Translate, or as part of larger systems, such as providing medical information to communities that speak low resource languages [38]. Because of the wide swathe of possible applications, it is important for us to make systems that are fair to all of their users.

In this work we will study the relationships between two bias metrics when using the Spanish language. One of our metrics is intrinsic, that is, it measures bias found in word embeddings. These are representations of the words that encode characteristics of the language. The other metric is extrinsic, that is, it measures bias found in downstream tasks, in our case this task will be hate speech detection. We will be using two different embedding algorithms to be able to properly compare our results.

Throughout our experiments we look to determine how reliable the relationship between these two metrics is and whether we should trust that changes in one will reflect on the other in a similar way. We reach the conclusion that the relationship is unreliable and that there are some issues with how our intrinsic bias metric was designed to be used with Spanish.

In chapter 2, we give a quick overview of what bias is, how it affects society and how it reflects in machine learning systems. In chapter 3, we explain the different

concepts that we work with and how they interact with each other. In chapter 4, we talk about our experiments and analyze their results. Finally, in chapter 5, we connect our findings to the existing literature and mention several ways in which our work could be expanded upon.

We should note that due to the nature of our downstream task (hate speech detection), some of the information in the background section might be disturbing to some readers. We will also talk about insults and slurs in Spanish in section 3.3 in order to explain some of the choices we made when preprocessing our data.

Chapter 2

Background

In this chapter we will talk about bias, discrimination, and fairness, both from a social and from a technical standpoint. We will talk about the former in section 2.1, while the latter will be considered in section 2.2.

2.1 Motivation

2.1.1 Bias and Discrimination

Biases in abstract are preferences or tendencies in favor or against someone or something. However, there is a stark difference between biases that originate from personal preference (i.e. liking a given ice cream flavor or not) to those that generate discrimination and prejudice. While these might seem like just ideas or opinions, they do tend to have an actual impact in the world.

An example of this is the police brutality towards people of color in America [15], which has lead to several murders of Black people. One of the most recent being that of George Floyd in May 2020, which revitalized the Black Lives Matter movement [33] and also led to a series of protests and riots across America.

These kinds of biases also find their way into machine learning systems. For example, there have been attempts to use data of previous arrests in order to create profiling systems that try to predict whether someone is likely to commit a crime or whether they should be considered suspects in a case. Given the amount of bias that police tends to show towards people of color [42], it is unsurprising that these kinds of systems are less accurate with these groups, sometimes showing an alarming amount of false positives [28].

Another example is that hate speech detection systems on tweets tend to perform much worse when dealing with African American English than with Standard American English [11]. Blodgett et. al. [5] note that these kinds of biases in machine learning systems help to normalize and perpetuate systemic, historical, and discriminatory practices.

2.1.2 Why are We Focusing on Bias Against Women?

While there have been feminist movements in Latin America since the late 60's, it is still one of the regions with the highest rates of feminicides. Mexico and Brazil lead with the most in the region, with countries such as Honduras, El Salvador, and Guatemala having higher per capita rates [17]. Because of this, numerous feminist movements have been recently appearing throughout the region.

In the last few years, there has been a constant malcontent in Chile towards the government because of several of their policies. One of the major issues was the lack of recognition of the issues that affected women in the country. The student demonstrations that ensued in 2018 were closely related to some feminist collectives [2], which created the performance "A Rapist in Your Path", which includes the chant "And it's not my fault, not where I was, not how I dressed. And the rapist was you". This performance has become an international anthem, with performances being made throughout the world [29].

From that year on, there have been increasing feminist demonstrations throughout the region, most of which have been met with mockery and constant hate speech in social media. Two of the main movements have been "Ni Una Menos"[1] throughout Spanish-speaking countries against the soaring rates of feminicides in recent years [39] and the "Não é Não"[2] in Brazil due to the lack of legislation and acknowledgement regarding consent [22].

These movements culminated in widespread strikes in the region before dying out due to the current pandemic [25]. However, reports of domestic violence and feminicide have greatly increased during the quarantine period [10]. Because of that, there were several feminist demonstrations in Turkey in July which closely echo the motives and the imagery used in the ones in Latin America [30].

[1] Not One Woman Less
[2] No Means No

2.1.3 Why are We Focusing on Bias Against Migrants?

In general, mass migration movements tend to arise out of need, be it escaping from violence, poor economic or social environments or natural disasters [23]. However, they tend to be met with distrust and discrimination at best and with governments that have set oppressive mechanisms in order to stop them at all costs at worst [46].

In recent years, there have been three main migration phenomena in Spanish-speaking countries.

The first of these is from Central America and Mexico to America . This movement started in the 80's and, while there has been ebb and flow, the stream of people has been constant. One of the recent surges came as a massive caravan from Honduras to America [1], picking more people along the way. These people were constant targets of violent criminal organizations, including the Mexican drug cartels and criminal gangs. The caravan was promptly stopped when it reached the border between Mexico and America, where immigration agents from both countries held the migrants in detention centers [41, 40], which often have inhumane living conditions.

Another major migratory movement that also happened was from Venezuela to the surrounding countries after the current president, Nicolás Maduro, came into power [45]. In 2018, after major political reforms and a deep economic crisis and famine, an even greater amount of people decided to leave their home country. This movement increased drastically after the government decided to use force to disperse demonstrations [8].

The third major movement is from the north of Africa and South America to Spain. This represents another constant stream of people and recently got another influx of people after the Algerian protests of 2019 [35]. This is considered one of the most perilous migratory routes due to the dangers of crossing the Strait of Gibraltar [19].

All of these movements have been accompanied by fear mongering and dehumanization of the migrants, both from politicians and from the people at large [46]. This is reflected in social media through messages that tend to justify the hardships that the migrants have gone through or to blame issues from the target country on them [44].

2.2 Technical Background

Throughout this section we will explain some of the technical concepts that we will be dealing with and how they relate to bias and discrimination.

2.2.1 Word Embeddings

Word embeddings are one of the central parts for the current state of research in NLP. They allow us to encode and feed syntactic and grammatical properties of words to machine learning systems without having to engineer them by hand. In this section we will give a quick introduction to word embeddings and what they are.

Computers cannot understand words and what they mean the same way as we do. If we give them a word such as ”pizza” or ”taco”, they will only notice that we gave them a string of characters. One way to make them notice what words are is to compile a vocabulary[3] from our corpus. We then transform the tokens into vectors where each entry corresponds to a different word from the vocabulary and there is a zero at all but the one corresponding to the current word. This is called one-hot encoding.

Even though we have succeeded in telling computers what words are, we still run into two big issues. The first one is that languages tend to have tens of thousands of words, so we end up having enormous sparse matrices, especially when our input text is large. This means that we end up using too much memory and take a lot of time to make each operation. On the other hand, these representations tell us nothing about the individual words. If we take the vectors corresponding to ”cat” and ”dog”, they will have the same relation between them as ”sushi” and ”car” do. That’s where word embeddings come into play.

We know that the parameters in the hidden layers of neural networks tend to encode latent information from our input data. Word embeddings take advantage of these intermediate representations to assign smaller, dense vectors to each token while including some of its semantic and syntactic information. Depending on how we use the neural network models, we can have non-contextual and contextual embeddings.

Non-contextual embeddings are static representations that are usually taken from more shallow networks. These assign a dense vector to each word type in the vocabulary. One of the more desirable properties from this kind of embeddings is that we can easily interpret some of the grammatical information encoded in them. For example, in the original word2vec paper [31], they showed that if you make the operation ”king” - ”man” + ”woman” with the corresponding vectors, you get the one that represents ”queen”. Similarly, if you take the vectors for a country and add ”france” - ”paris”, you will get that country’s capital city. However, as we will discuss further in section 2.2.3, this can lead to non-desirable outcomes when bias is involved. Another down-

[3] A list of the words that appear in our data. We can always substitute rare words with an ”unknown” token to keep our vocabulary’s size in check.

side of this kind of embeddings is that, as they assign a single, static vector to each type, they do not take into account the context in which each token appears, which is especially noticeable when dealing with tasks that require so. Three of the most commonly used algorithms are word2vec [31], GloVe [34], and FastText [6].

On the other hand, contextual word embeddings are obtained using deep neural networks. Here, instead of extracting an intermediate representation, we pre-train the network and freeze most of the weights, except the ones in a few of the outer layers. We then substitute the output layer with the network that we will be using for our task and train that and the non-frozen layers of our embeddings network. Here we are gaining the ability to recognize the context in which tokens appear in, but are losing the interpretability that non-contextual embeddings had. Two of the more used ones are BERT [12] and ELMo [37]. However, it is important to note that these networks require much more resources to train than those used for non-contextual embeddings [43].

Given that non-contextual embeddings are more easily interpretable and no one has as of yet studied the effects that altering intrinsic bias has on the extrinsic bias, we consider that focusing on this kind of embeddings would be a good place to start. Furthermore, our instrinsic bias metric, XWEAT, was created with non-contextual embeddings in mind, so mixing both kinds would lead to us having to use two different intrinsic metric, which in turn would lead to our results not being as valid as if we had used the same intrinsic metric for all of the embedding algorithms we will use. This leaves the door open for future research on this topic to be made using contextual word embeddings.

2.2.2 Bias in NLP

While the use of embeddings and neural networks have greatly advanced the state of the art in NLP research, they have also brought along most of the issues and concerns that tend to accompany these kind of systems.

One of the major concerns is that neural networks in general tend to act like black boxes due to their lack of interpretability. This leads to us not being able to properly identify potential issues early on, such as unfairness. Furthermore, we know that machine learning systems in general tend to pick up and amplify bias found in the training data, either from the actual content of the data [13] or due to biases from the people annotating it [48].

One example are correference systems, which aim to identify the word to which a given pronoun is referring. These systems tend to replicate gender stereotypes, such as assigning mostly male pronouns to words such as ”doctor” and female pronouns to words such as ”nurse”, even if it is clear from the rest of the sentence that it should not be the case [50].

An example related to hate speech detection would be in the paper by Dixon et. al. [13]. They noticed systems designed to flag toxic content on Wikipedia would often classify any message containing the word ”gay” as toxic. This was due to that word being severely underrepresented in non-toxic comments as a gender identity marker and appearing mostly in toxic comments as an insult.

One place where these biases have been found on machine learning systems are word embeddings, more specifically, non-contextual ones. As we mentioned in section 2.2.1, these have amazing properties such as being able to do arithmetic with the vectors corresponding to the tokens. One of the examples that we mentioned was that if we take ”king” and add ”woman” - ”man”, we get the vector corresponding to ”queen”. However, these also reflect biases. For example, Bolubaski et. al. [7] noticed that if we take the vector for ”computer programmer” and do the same operation, we end up with the vector for ”homemaker”!

2.2.3 Reducing and Removing Bias from Word Embeddings

Bias can have unintended and undesirable consequences in our systems, both from the technical and the human standpoints. Because of this, several attempts to remove bias from word embeddings have been made.

In their paper, Bolubaski et. al. use PCA, a method through which we linearly transform a space so that the dimensions that carry the most information are aligned with the axes, to identify and remove the dimensions that serve to identify a given group [7]. Another proposal has been to use dataset balancing so that no group is over represented [13] or so that certain words appear in more varied scenarios [50].

However, removing bias is not as straightforward as it might seem at first glance. Wang et. al. [47] show that these biases are deeply rooted in our cultures and languages. Therefore, they tend to be highly interdependent with other apparently unrelated variables. The end result is that instead of removing the biases, we just end up hiding them. One way to see this is by visualizing word clusters. If we go just by visual analysis, we will not find the clusters anymore. However, if we color each cluster,

we will see that the two groups have not mixed with each other, even after running the algorithm to reduce bias.

It is important to note that most research in these topics has been focused on the English language and on reducing the intrinsic bias on word embeddings. However, as noted by [5], no one has as of yet studied the correlation between intrinsic bias in word embeddings and extrinsic bias in a downstream task. We will be focusing on how altering bias on the embeddings affects bias on hate speech classification.

For our work, we will be using Spanish, a romance language. As with the rest of the languages in that family, it has gendered nouns and adjectives. It is important to note that in languages with grammatical gender, words tend to cluster around that rather than semantic similarity. We chose this language as it would allow us to analyze how altering the intrinsic bias of the embeddings will affect gender bias as compared to other kinds of bias.

2.2.4 Fairness and Equality of Opportunity

Depending on how the hate speech detection models are usually deployed to flag potentially hateful or abusive messages on social media. Therefore, the kinds of bias that we will be studying in this paper can cause allocative harm. That is, it can cause resources to be distributed in an unfair manner. In our case, it could mean that innocuous messages get incorrectly flagged as hateful or that hateful messages are ignored due to them being incorrectly classified. We gave an example of this in section 2.2.2.

One idea that springs to mind to solve this is to use what is called demographic parity. That is, to make it so that a decision is independent of the protected characteristics. However, this notion of fairness is flawed [14], as it does not necessarily mean that it will give the same opportunity to similar individuals, especially when the data is heavily unbalanced.

One thing that Hardt et. al. [21] note is that the burden of ensuring fairness should be on the model rather than on the different groups. They propose a metric based on equal opportunity to access to resources, of which we will talk more in section 3.4.

While reducing bias is a very important endeavor, we should note that fairness is what actually reduces discrimination, at least in technical settings.

Chapter 3

Experimental Setup

In this chapter we will talk about the different parts that make up our pipeline and how they relate to each other. We will begin by explaining which word embedding algorithms we will be using in section 3.1. We will then describe our downstream task, hate speech detection, in section 3.3, along with the dataset and the network that we will use for it. In section 3.4 we will talk about both of our bias metrics and in section 3.5 about how we will be modifying the geometry of the embedding space. Finally, we will describe how these different parts slot into our pipeline in chapter 4.1.

3.1 Word Embeddings

As mentioned in section 2.2.1, word embeddings are a way to encode syntactic and semantic information of words into vectors. For our work, we will be focusing on the word2vec and the FastText embedding algorithms.

It is also important to note that Gonen et. al. [18] showed that in Italian words tend to cluster around grammatical gender, rather than around semantic similarity. Given that Spanish and Italian are both romance languages with grammatical gender, we would expect a similar effect. This will most likely have an effect on how our classification system interacts with hate speech against women when compared to hate speech against migrants, as we will explain further in section 3.3.

3.1.1 Word2vec

The word2vec algorithm was proposed in 2013 by Mikolov et. al. [31]. There are two machine learning models that can be used to obtain these embeddings: CBOW and

skip-gram.

In the CBOW[1] model, we try to make our model predict the current word given a window of n words both before and after the target. On the other hand, the skip-gram model takes a word and tries to predict the surrounding words within a given window. Both models take a corpus of text and output the weight matrix of their single hidden layer. This weight matrix is the word embeddings, and is then used to pass the sparse one-hot vectors that represent each token to dense ones.

For our project, we chose the skip-gram algorithm, as it is one of the most widely used. More specifically, we will be using the skip-gram model as implemented in the gensim python package [52]. We will also use 300-dimensional vectors, as that is one of the standard embedding sizes that is currently used. All of the other hyperparameters of the model will be the default ones given by the package.

3.1.2 FastText

The FastText embedding algorithm was proposed in 2016 by Bojanowski et. al. [6]. It seeks to enrich the methods used in word2vec through the use of subword information. This algorithm was published by facebook[2] and is written in C++ for faster training, hence the name.

One of the issues with word2vec, as mentioned in section 2.2.1, is that word embeddings tend to cluster around grammatical variations such as gender. This algorithm was designed to diminish this effect, while also being able to interact with out of vocabulary words. It learns the representations of the n-grams, subsets of n contiguous letters that form a given word. For example, the bigrams for ”pizza” would be ”pi”, ”iz”, ”zz” and ”za”, while the trigrams would be ”piz”, ”izz” and ”zza”. Then, the vector corresponding to that word is obtained by averaging the representations for those embeddings. This is done under the assumption that morphological variants will only affect a small subset of the letters that form a word. Thus, we would expect the embeddings to cluster more closely around meaning than around things such as grammatical gender or number in the case of Spanish. On the negative side of things, this algorithm is likely to consider that ”cat” and ”car” are much closer in meaning that they have any right to be.

Given that most nouns and adjectives in Spanish have these two characteristics, we chose to use the FastText algorithm for our project. We consider that using subword

[1] continuous bag of words

[2] https://fasttext.cc/

information should allow us to gain more insight into how the representation of grammatical gender affects how our model classifies hate speech directed towards women, as opposed to how it performs when dealing with hate speech against migrants.

For our project, we will be using the re-implementation of this algorithm in the gensim python package[52]. As in the case of word2vec, we will be using skip-gram 300-dimensional vectors, with all of the other hyperparameters being the default ones.

3.2 Dataset for the Embeddings

Twitter offers several data streams through their API, the smallest being the "Spritzer" stream [27]. In order to train our embeddings, we used data from the archive.org[3] backup of this stream from March 2019. Given that the dataset used for our downstream task was also released on 2019, we believe it to be a good representative of what the words meant around that time.

We used the language identifier in the tweet data to filter out the ones that were not in Spanish. We also removed retweets so that we would not have duplicated data. Not doing this might have had a noticeable effect, especially when dealing with tweets from famous people or those that have rare words. In the end we were left with a little over 1.7 million tweets, which had over 10 million tokens.

We preprocessed the tweets by removing URLs, mentions, and hashtags and replacing them with the <URL>, <MENTION>, and <HASH> tokens, respectively. We also substituted words that appeared less than ten times with the <UNK> token. This was because words that appear that sporadically are most likely either uncommon typos or rare words. Either way, having a type appear less than that would generate an embedding which would contain no meaningful information and would probably end up being the same or worse than a random embedding. We also turned all of the words to lowercase and got rid of all of the extra spaces and linebreaks within the tweets.

Stemming is a method that is sometimes used to remove grammatical variations of words[4]. Depending on the downstream task, this can be either a good or a bad idea. Given that part of ours is detecting hate speech against women, we consider that keeping grammatical gender of words will help the performance of the network we use in the downstream task. This will be explained further in section 3.3. We are also

[3] https://archive.org/details/archiveteam-twitter-stream-2019-03

[4] For example, in English, if we stem "play", "plays", and "played", we get "play". This way, grammatical variations do not get in the way of the core meaning of the word. On the other hand, some nuances of the language are lost.

expecting that this could potentially affect how our two embedding algorithms will react to modifying the intrinsic bias.

3.3 Downstream Task

3.3.1 Hate Speech Detection

Social media has completely transformed how people communicate with each other. We can easily broadcast messages to hundreds or thousands of people, without leaving our couches and in complete anonymity. However, because of this, people are able to say things that they would not normally say face to face, such as hateful messages. The Council of Europe notes [16] that hate speech can lead from the normalization of these kind of ideas to physical aggression to even greater acts of violence, such as genocide.

Some social media platforms have filters intended to decrease hateful and abusive messages from their platforms. However, there are a lot of messages that still manage to bypass the filters [36]. As mentioned in the first example of section 2.1, the existence of intrinsic bias relating to this kinds of systems can lead to big groups of innocuous messages being incorrectly misclassified or hateful messages being let through. Meanwhile, on our second example we were able to see how these kinds of intrinsic biases can lead to unfairness when comparing the performance on two different groups.

As we will be comparing extrinsic bias from hate speech against women and hate speech against migrants in Spanish tweets, it is important to know that the same word can have completely different meanings when in masculine and femenine forms. Take for example ”dog”. In its masculine form, ”perro”, it is almost used to describe the animal. When used to describe people, it means that the person is despicable or lazy. On the other hand, the femenine form ”perra” is often used in the same kind of contexts in which one would use the word ”bitch” in English as an insult and less commonly to describe a female dog. A more extreme case would be the word ”puto”. When used with humans, this word is a slur against homosexual people. On the other hand, it’s female version, ”puta”, is a slur against women that implies sexual promiscuity.[5]

On the other hand, insults against migrants tend to have gender variations for both grammatical genders, such as the slur ”beaner”, which is translated to ”frijolero” and

[5]It has been hotly debated whether the use of this word on inanimate objects or as interjections is related to the slurs and how much. When a euro-centric, presciptive linguistic organization from the Spanish government chimed in to say that most of these usages were slurs, accusations of erasure of dialects from Latin America and of cultural imperialism began to flare. To the knowledge of the author of this work, the debates have been mostly inconclussive.

”frijolera”, both of which are still slurs against Mexican migrants to America. On the other hand, we have slurs such as ”sudaca”, used in Spain against South Americans, which is gender neutral, despite having female grammatical gender.

This should help us to be able to see the effects of gender bias versus topical bias in our extrinsic metric.

3.3.2 HatEval Datset

For our hate speech detection task, we will be using the HatEval dataset [3]. It was introduced as task 5 of the 2019 SemEval workshop. This task consisted on detecting and clasifying hate speech in Spanish and English both against women and against migrants. For this work, we will only focus on the Spanish dataset.

The Spanish HatEval dataset is divided into training, validation, and test sets. These sets consist of $4,500$, 500, and $1,600$ tweets, respectively. All of the sets have a mixture of hate speech against women (which we will be calling Group 1) and of hate speech against migrants (Group 2), with the test set having an equal amount of tweets for each group.

The task is further divided into two subtasks, depending on how fine-grained the classification is. Subtask A is a binary classification problem where each tweet is classified as either hate speech or not. On the other hand, Subtask B is a multiclass classification problem, where we must classify whether a tweet is hate speech or not and, if it is, whether it is targeted at an individual or not and if it is aggressive or not.

3.3.3 Evaluation Metrics

In order to define our evaluation metrics, we must first define a couple of terms.

A *true positive* (TP) is when our model correctly predicts that a tweet has a given label. For example, in the case of task A, it would be when a hateful tweet gets classified as such. Similarly, a *true negative* (TN) is when our system correctly predicts that a tweet does not have the label. In the case of task A, it would be classifying a tweet that is not hate speech as such.

On the other hand, a *false positive* (FP) is when a tweet is incorrectly given a label. In line with the examples above, it would be when our system labels a tweet as hate speech, even when it is not. Finally, a *false negative* is when our system incorrectly predicts that a tweet does not have the label. Here, it would be when a tweet is classified as not being hate speech, even though it is.

With these definitions, we can finally talk about our evaluation metrics. The ones that we will report will be the following:

- **Accuracy** - the rate of correct answers. This is the metric we will use to determine the best model and whether to do early stopping.

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

- **Precision** - the ratio of true positives to predicted positives. Basically, if our system assigns a label to a tweet, this is the probability of that prediction to be accurate.

$$\text{Pr} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

- **Recall** - the ratio of true positives to tweets that have the label. Basically, if a tweet has a label, this is the probability that it was correctly classified.

$$\text{Rc} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

- **F1-Score** - the harmonic mean of the precision and recall. It is meant to give us a general idea of how our model is performing.

$$\text{F1} = 2\frac{\text{Pr} \cdot \text{Rc}}{\text{Pr} + \text{Rc}}$$

There are two other metrics to consider for task B. They are obtained by evaluating each class separately and obtaining the metrics mentioned above.

- **Partial Match** - a summary of the metrics for each separate class is given by the average of their F1-scores. For example, this would account for a hateful and agressive but not targeted tweet that was classified as being all three.

$$\text{F1}_{\text{Task B}} = \frac{\text{F1}_{\text{Hate Speech}} + \text{F1}_{\text{Targeted}} + \text{F1}_{\text{Agressive}}}{3}$$

- **Exact Match Ratio** - the rate of predictions that exactly matched the labels. With the previous example, it would only take into account that tweet if it was correctly classified. If even one of the labels is off, it would not consider it.
  In mathematical terms, if Y_i is our target label for tweet i and P_i our predicted labels, let $I(Y_i, P_i)$ be 1 if $Y_i = P_i$ and 0 otherwise. Then, the exact match ratio (EMR) is given by:

$$\text{EMR} = \frac{1}{|\{\text{set of tweets}\}|} \sum_{i \in \{\text{set of tweets}\}} I(Y_i, P_i)$$

3.3.4 Convolutional Neural Networks

For both of our classification tasks, we will be using the convolutional neural network (CNN) proposed by Kim [24].

Our CNN model consists of the following layers:

- **Input Layer** - before feeding our data to the network, we must transform the words in each tweet into their respective embeddings. Therefore, we get an $n_a \times 300$ input matrix for tweet a, where n_a is the number of words in that tweet.
- **Convolutional Layer** - here we take several kernels and use them for convolutions with our input matrix.
  - Kernels are matrices whose values are learnt during the training of the network. A kernel b of size m_b is a matrix of $m_b \times 300$.
  - A convolution is a windowed, weighted average of the input matrix, using the kernel to indicate the weights of the average and the size of the window. Mathematically, if we have a convolution of input matrix a and kernel b, the output of this layer would be a feature map defined by

    $$(b * a)_i = b \cdot a_{i:i+n_a-1},$$

    where $a_i : j$ is the submatrix of a that goes from row i to row j, for $i < j$. This process can be visualized in figure 3.1.
  - Our model will have two filters of each of the following sizes: 2, 3, and 4. Therefore, the output of this layer will be six feature maps, one for each kernel.
  - A non-linear activation function will help our model learn.
- **Pooling Layer** - we take the maximum value of each feature map and concatenate the results. This is done so that we take the most important feature from each map, while getting rid of the issue of tweets being of variable lengths.
- **Linear Layer** - here we multiply the output of the previous layer by a learnable matrix and add a learnable bias term so that we get an entry for each label in our classification problem. Given that all of our classification problems would be binary, we get a single number from this layer for each tweet. We then normalize this number using a sigmoid function and get the probability that our tweet is in a given class or not.

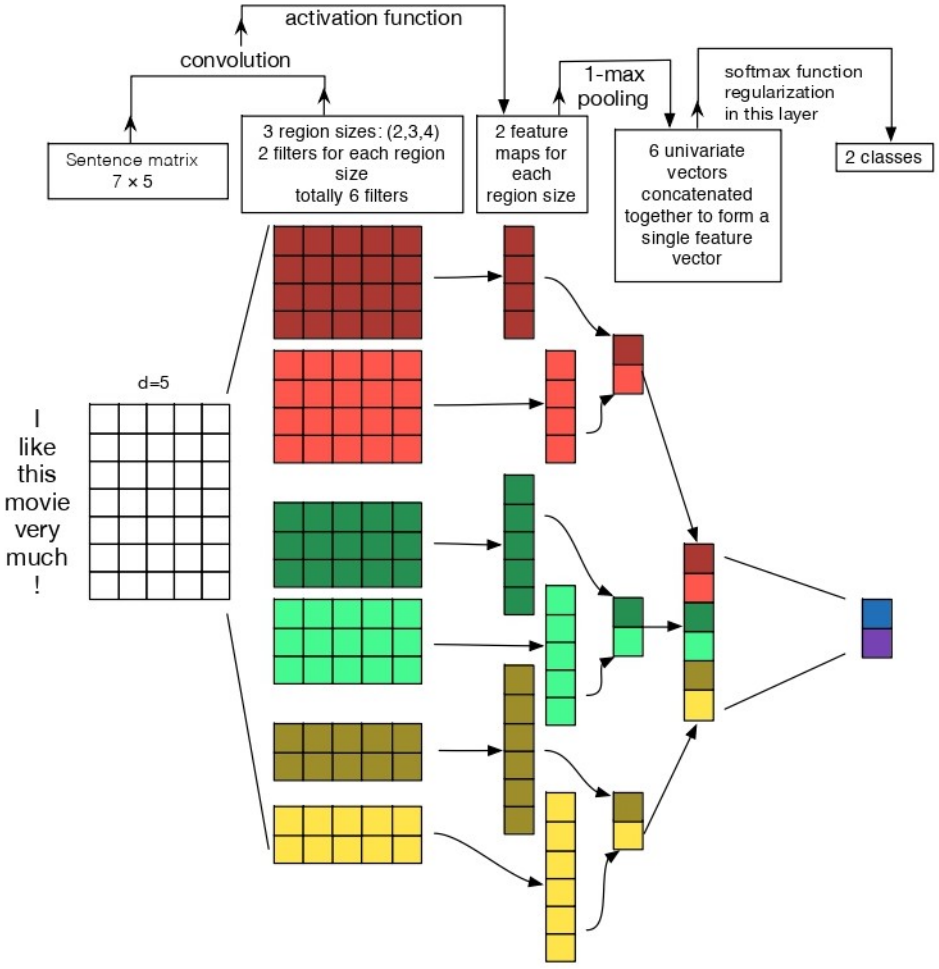

Figure 3.1: Diagram of the CNN architecture. Taken from Zhang and Wallace [49].

For subtask A, we will use a single network to determine whether a tweet is hateful or not, while for subtask B, we will use a different network for each of our classes. This is similar to the proposal of the highest scoring team [4]. They proposed to use three different classifiers, one for each label and trained separately.

3.4 Bias Measurements

In order to measure bias on the embeddings and its effects on bias on downstream tasks, we will be using two different metrics. In order measure intrinsic bias (i.e. bias as reflected in the word embeddings), we will be using the XWEAT tests, while we will use equality of opportunity to measure extrinsic bias (i.e. bias as reflected in a downstream task). We will explain both of these more in detail in the following sections.

3.4.1 Word Embedding Association Test

The word embedding association test (WEAT) [9] is composed of a series of tests meant to mirror those used with humans, such as Harvard's implicit association test (IAT) [20].

There are ten tests in total, which are further explained in table 3.1. Each test has two lists of targets and two lists of attributes. The sets of targets are the lists between which we want to measure bias. Therefore, if we are measuring gender bias, these could be stereotipically male names on one list and stereotypically female names on the other. The sets of attributes are the lists towards which the bias will be measured. In the case of gender bias, this could be a list of professions that are stereotipically male and a list of professions that are stereotipically female.

While these tests were originally released in English, Lauscher and Glavaš [26] released XWEAT, which added multilingual and cross-lingual support to the original WEAT. The languages supported are German, Spanish, Italian, Croatian, Russian, and Turkish. In order to obtain the lists for each test, they used automatic translation and then asked native speakers to fix incorrect translations and to propose more fitting ones. We do not consider this to be an effective use of resources, neither from an ecological nor a human standpoint, especially as neither of the lists are particularly large. They also accounted for grammatical gender in their translations. That is, if one of the words in English has grammatical gender in the language that they are translating to, all of the

gender variations of that word are included in the corresponding translated list. These things lead to some things that could potentially become issues during our experiments, some of which we will mention at the end of this section.

For our experiments, we will be considering Tests 1, 2, and 6 through 9, as suggested by the authors of the paper. The topics covered in the targets and attributes lists can be found in table 3.1, with a full list of the attributes and targets of Test 7 can be found as an examlpe in section 4.3 in table 4.4.

	Target Sets		Attribute Sets	
Test 1	Flowers	Insects	Pleasant	Unpleasant
Test 2	Instruments	Weapons	Pleasant	Unpleasant
Test 3	Euro-American names	Afro-American names	Pleasant	Unpleasant
Test 4	Euro-American names	Afro-American names	Pleasant	Unpleasant
Test 5	Euro-American names	Afro-American names	Pleasant	Unpleasant
Test 6	American Male names	American Female names	Career	Family
Test 7	Math	Arts	Male	Female
Test 8	Science	Arts	Male	Female
Test 9	Physical condition	Mental condition	Long-term	Short-term
Test 10	Older names	Younger names	Pleasant	Unpleasant

Table 3.1: Contents of the different XWEAT target and attribute sets. This table is based on that from [26].

The tests have three relevant values attached to each of them:

- **WEAT Statistic** - compares the average similarity between the sets of targets and the sets of attributes. The higher this number is, the more bias is present in the embeddings for that test.
- **Effect Size** - this is a normalized metric that compares the association distributions for each target set. It can be considered as the amount of bias present in the embeddings, so this will be the value we will be focusing on.
- **p-value** - this is the probability that the observed bias was due to randomness instead of actual bias in the embeddings.

Considering that none of the tests expiciltly refer to concepts relating to migration, we consider that modifying the bias on these tests will have a much greater impact on group 1 (hate speech against women) from the hate speech detection task than on

group 2 (hate speech against migrants). Another variation of the WEAT tests has been proposed for the Spanish language [51], however, we chose not to use those as they only consider gender bias and their authors note that it is meant to be used to recognize bias and not to attempt to reduce it.

For example, some of the words are broken up into more than one, even if a single word variation also exists. An example for this is ”hopeless” in Test 8, which is translated as ”sin esperanza”, when ”desesperado” or ”desesperanzado” have a similar, but admittedly different, meaning. There are also words that have been translated with a certain bias. For example ”hers” and ”his” in Test 7 can both be translated as either the gender neutral ”su” or ”suyo”/”suya” or as the two-word gendered versions ”de él” for ”his” and ”de ella” for ”hers”. However, they translated ”his” as ”su” and ”hers” as ”de ella” and ”su”. On this same test they translated ”male” as ”masculino”, which tends to mean either the grammatical meaning of the word or ”manly”, while they translated ”female” as ”hembra”, which tends to mean ”female” in a biological way (that is not usually applied to humans).

One of the things that the creators of XWEAT note is that some of the tests were made considering the biases relating to American names. Because of this, they do not recommend Tests 3 to 5 and 10 to be run on non-english word embeddings. They consider that the names present in Test 6 should appear often enough in the other languages' embeddings. Regardless of this, we consider that not translating the names is an oversight that goes on to show that the creators of XWEAT might not have considered that different languages are immersed in different cultural contexts to each other.

3.4.2 Equality of Opportunity

Equality of opportunity is a metric meant to measure how fair is an NLP system. The idea is that if we separate two groups according to their protected characteristics, our model should avoid discrimination based those characteristics. In our case, the protected characteristics will be gender for our first group of the HatEval data and being a migrant, for our second group.

Given classes C_1, C_2, we say that our model satisfies equality of opportunity if and only if $\mathrm{Rc}_{C_1} = \mathrm{Rc}_{C_2}$. That is, both classes have the same opportunity of being classified correctly. Because of this, our equality of opportunity metric will be defined as:

$$\text{Eq. of Opp.} = \mathrm{Rc}_{C_1} - \mathrm{Rc}_{C_2}$$

Another related metric that we wil be using would be the difference between the

precisions. This basically tells us if both classes that were given an opportunity were classified correctly. This measure is given by:

$$\text{Precision Diff.} = \text{Rc}_{C_1} - \text{Rc}_{C_2}$$

Given that subtask A of our downstream task is a binary classification problem, calculating the equality of opportunity metric should be straightforward enough. For subtask B, we will be calculating the equality of opportunity metric for each label and report the average value, as with the F1 score. This will serve to give us a general idea on how much bias is present in the system across all labels.

3.5 Altering Bias

For our work, we will directly modify the geometry of the embeddings in a way that we can predictably modify the amount of bias reflected in the XWEAT tests results. For this, we will be using the attract-repel method.

3.5.1 Attract-Repel

Attract-repel was originally proposed by Mrkšić et. al. [32] as an algorithm to improve semantic quality of word embeddings either in mono-lingual or in cross-lingual spaces.

This algorithm works by giving a list of pairs of synonyms (that is, that we want to be more similar) and a list of pairs of antonyms (words that we want to be farther apart). It the feeds the embeddings and both lists to a neural network and uses a loss function to measure how much to alter the embedding space. The loss function takes into account the following principles:

- The loss function should incentivize pairs of synonyms to be closer to each other.
- The loss function should incentivize pairs of antonyms to be farther apart from each other.
- A regularization term must be added so that the general semantic content of the embeddings is not modified (with the two exceptions mentioned previously).

For each of the XWEAT tests, we will be run an experiment where we either increase or decrease bias in the embedding space for that specific test. To increase bias, we will take the lists of targets and for each word in them, consider each attribute of

the stereotypical list to be a synonym and the ones in the antistereotypical list to be antonyms. On the other hand, we will reduce bias by making the targets synonyms with their antistereotypical attributes and antonyms with their stereotypical ones.

Chapter 4

Experiments

In this chapter we will begin by explaining how we ran our experiments in section 4.1. We will then give the results of our two bias metrics and their correlations in section 4.2. While not the focus of this work, we will also report the results from our downstream task and how it relates with the results of the XWEAT tests. This is to verify that altering instrinsic bias does not render our network useless. Finally, we will give an analysis of our results in section 4.3.

4.1 Methodology

Before running any experiments, we will preprocess the dataset for our embeddings as described in section 3.2. We will also preprocess the text of the HatEval tweets the same way. We will then generate the FastText and word2vec embeddings using this preprocessed data.

Once we have our embeddings, we will use the attract-repel algorithm as described in section 3.5. That is, for each of the XWEAT tests that we will use[1], we generate the synonym and antonym lists so that we can increase or decrease bias according to that test. We will then use these lists for each of our embeddings. The end result is that we will have 13 versions of the embeddings for each of our algorithms - the original embeddings and, for each of our six tests, one with increased and one with decreased bias. Of note is that we will be grouping our results on the embedding algorithm from which they came.

For each of our embeddings, we will run the XWEAT tests and train the CNNs for each of our tasks. Once we have the results from these, we will calculate the equality of

[1] We will be using XWEAT tests 1, 2, and 6-9. That is, six tests in total.

opportunity metric. We will report both the bias metrics and the performance metrics of our tasks, as well as the correlations between them.

We list the code that we will be using and the modifications we made to it in appendix A.6.

4.2 Results

In this section we will report the values that are the most relevant to us. In appendix 4.2, we will show the rest of the results from our experiments. This includes all of the values given by the XWEAT tests, the performance results for both of the subtasks of our downstream task and the results from our extrinsic bias metrics.

The distributions for the effect sizes of the XWEAT embeddings can be found on figures 4.1 and 4.2. We can see that most of the values tend to cluster near the one of the unaltered embedding, save for a few outliers. These correspond to the embeddings where we modified the bias for a highly correlated test. As expected, the higher the effect size, the lower the p-value tends to be. Another thing to note is that on both types of embeddings, the results of Test 6 have no outliers, not even in the respective modified embedding and that most of the p-values for Test 8 are very high. We will comment on these and other details in section 4.3.

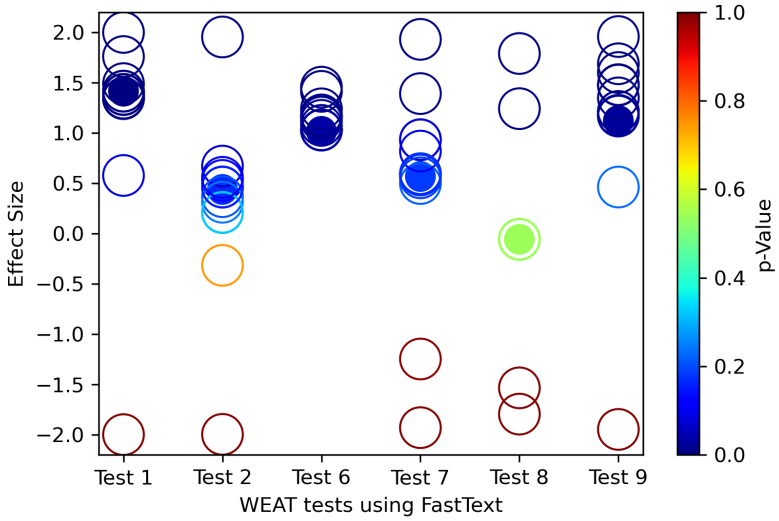

Figure 4.1: Distribution of the XWEAT effect sizes for each test when using FastText embeddings. The color corresponds to the p-value of each data point and the solid circle corresponds to the unaltered embedding.

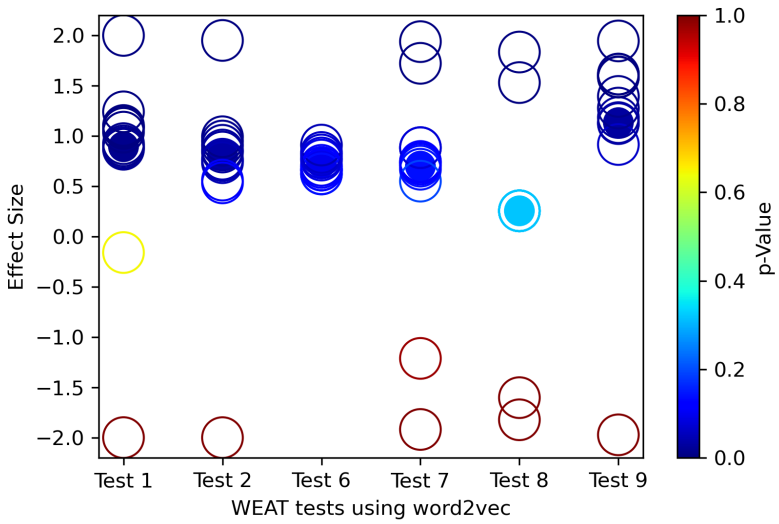

Figure 4.2: Distribution of the XWEAT effect sizes for each test when using word2vec embeddings. The color corresponds to the p-value of each data point and the solid circle corresponds to the unaltered embedding.

We can see the results from our unmodified embeddings in table 4.1. For subtask A, there were no noticeable changes with either embedding in any of the metrics, with most values being between 0.76 and 0.75. For FastText, the unmodified embedding was the best performing under all characteristics and the lowest F1-Score was of 0.73. Meanhwile, for word2vec we did get improvements for some of the metrics, but these were lower than 5%, so they should not have a big impact on the performance. The lowest performing of the word2vec embeddings was the one where we increased bias on Test 9 (physical and mental conditions compared to long and short term), where the F1-Score dropped to 7.1.

On the other hand, for task B we got that for both embeddings the aggregated F1-Score didn't change much either for any of the embeddings, with a variation of ± 0.02. On the other hand, the EMR value did show a lot of fluctuation, with variations of ± 0.04. However, there are no strong correlations with any of the XWEAT tests for either of these values.

Finally, we can see the correlations between the XWEAT tests and the equality of opportunity metrics in tables 4.2 and 4.3 for FastText and word2vec, respectively. We used the Spearman correlation as we have no reason to expect the XWEAT test results to be linearly correlated to the equality of opportunity values. For subtask B, we will be reporting the equality of opportunity and the difference between precisions

	Subask A			Subtask B	
	Precision	Recall	F1-Score	EMR	Macro F1-Score
FastText	0.761	0.759	0.759	0.545	0.794
word2vec	0.756	0.753	0.754	0.508	0.773

Table 4.1: Results from our neural network for the HatEval task. Here we are only reporting the values for the unaltered embeddings. For the results of the altered embeddings, check appendix 4.2.

for each of the labels. Of note is that the, the equality of opportunity metric and the difference between recalls for hate speech detection was very high, so the correlation of the other variables will be statistically insignificant. We will talk more about this and other details in section 4.3.

4.3 Analysis

4.3.1 XWEAT Values and Statistical Significance

First we will talk about our XWEAT test results and their statistical significance. We will then talk about the equality of opportunity values, along with their correlations with the XWEAT test results.

We will first take a look at the different XWEAT tests, their effect sizes and their statistical significance. It it is important to note that in general higher effect sizes are strongly correlated to lover p-values. That is, the more bias there is, the less likely it is to have come from pure noise. A small summary of the statistical significance of these results can be found on table 4.5.

For our experiments, we ran one for each of our relevant XWEAT tests where we would use attract-repel to increase the bias for that specific test and one where we reduced it. After we got an embedding from each of these experiments, we ran all of our XWEAT tests on them. This gave us a datapoint of each test for each of the experiments that we ran. The following analyses take into account all of the datapoints for a given test.

- Tests 1 and 9 are our control groups. They are about plants and insects versus pleasant and unpleasant for Test 1 and about physical and mental conditions

		Subtask A		Subtask B						
				Hate Speech	Targeted		aggressiveness		Macro Average	
		Equality of Opportunity	Precision Difference	Equality of Opportunity	Equality of Opportunity	Precision Difference	Equality of Opportunity	Precision Difference	Equality of Opportunity	Precision Difference
XWEAT Effect Sice	Test 1	-0.28	-0.17	0.10	0.34	0.65	0.34	0.30	0.57	0.68
	Test 2 *	-0.28	-0.50	0.27	0.51	0.37	0.06	0.22	0.36	0.55
	Test 6	0.23	0.32	0.01	0.52	0.27	-0.36	0.17	-0.09	0.23
	Test 7 **	-0.33	-0.10	-0.30	-0.23	0.12	0.01	-0.54	0.01	-0.21
	Test 8 **	-0.22	-0.07	-0.38	-0.58	-0.27	0.15	-0.53	-0.03	-0.45
	Test 9	-0.29	-0.02	0.20	-0.02	0.07	0.08	-0.41	-0.09	-0.14

Table 4.2: Correlation between the XWEAT effect sizes and the equality of opportunity metrics when using FastText embeddings across all of our experiments. Rows marked with * denotes that there is low statistical significance and those marked with ** mean that the results are statistically insignificant.

		Subtask A		Subtask B						
				Hate Speech	Targeted		aggressiveness		Macro Average	
		Equality of Opportunity	Precision Difference	Equality of Opportunity	Equality of Opportunity	Precision Difference	Equality of Opportunity	Precision Difference	Equality of Opportunity	Precision Difference
XWEAT Effect Sice	Test 1	-0.07	0.47	0.44	0.33	0.51	-0.17	-0.29	0.42	-0.23
	Test 2	-0.23	-0.34	-0.03	-0.25	0.03	-0.06	-0.32	0.02	-0.27
	Test 6 *	-0.54	0.01	0.22	0.39	0.15	0.11	0.21	0.30	0.14
	Test 7 **	0.23	0.13	0.24	0.19	0.13	-0.21	-0.13	0.08	-0.25
	Test 8 **	-0.01	-0.06	0.45	0.43	0.48	-0.13	-0.14	0.33	-0.18
	Test 9	-0.04	-0.32	0.41	0.40	0.25	-0.14	-0.18	0.36	-0.24

Table 4.3: Correlation between the XWEAT effect sizes and the equality of opportunity metrics when using word2vec embeddings across all of our experiments. Rows marked with * denotes that there is low statistical significance and those marked with ** mean that the results are statistically insignificant.

versus being long and short term. None of them relate directly to either of the HatEval groups and have high statistical significance all over the board.

- Test 2 is instruments and weapons vs. pleasant and unpleasant. While the instruments list does not bear any relation with our downstream task, we expect that the amount of bias associated with the weapons should have an impact, as some hate messages involve threats of violence. This is especially notable when dealing with subtask B, which involves detecting whether hate messages are agressive and/or targeted to an individual or not. The p-values are low (< 0.03) for word2vec and the effect sizes are clustered near 0.75. This means that there is high bias and that it is statistically significant. On the other hand, the p-values for FastText are relatively high, between 0.3 and 0.15, save for the outliers, and the effect sizes are clustered around 0.5. This means that there is a low amount of bias, but it is not really statistically significant. This might be due to the cultural context in which these words appear. For example, most of the instruments are uncommon words, which would not be expected to consistently appear on tweets. Meanwhile, most of the terms for weapons range from those that are so old that I would expect them to be more closely related to fantasy settings to weapons currently in use. One way to think about this is that, while ”firearm” is a term that is currently in use in English, the term ”arma de fuego” sounds either too formal or the kind of term that would be used when talking of the 16th century, when the Spanish Empire set out to conquer the world.

- Test 6 compares stereotypically male and female names vs. work and family. These values did not really see much difference in variation, not even in the experiments specifically designed to alter bias here. This is most likely because the names are not stereotypical names from Spanish-speaking countries and would probably be more closely associated to celebrities and the like. Even though the creators of the translated XWEAT lists that we are using argue that this test should be useful in non-English languages, we believe that the results of this test point otherwise. The p-values for Test 6 are low (< 0.05) with effect sizes clustered around 1.25 for FastText. For word2vec, the p-values are higher (between 0.15 and 0.05) and the effect sizes cluster aound 0.75.

- Test 7 compares math and art versus male and female. It has p-values of around 0.15 ± 0.03 for both embedding types and effect sizes that cluster near 0.5. All except one of the terms in the art list are female, while half of the ones in the

math list are female. This can be seen in table 4.4. Considering that grammatical gender tends to overshadow topical characteristics, this could be the reason for this test to have very low statistic significance, save for the outliers. As mentioned before, Gonen et. al. [18] note that grammatical gender tends to have more impact on word embeddings than the actual meaning of the words. Because of that, we would expect that there will always be a more marked bias between the list of words related to art and the female words, regardless of the amount of topical bias present in the data used to train the embeddings.

- Test 8 compares science and art versus male and female. It has p-values of over 0.5 for FastText and over 0.3 for word2vec and other than the outliers, all of the effect sizes are essentially the same - 0 for FastText and 0.05 for word2vec. This is a test where we would have expected there to be a marked bias, so these results were surprising at first glance. However, all of the target words in both lists are female, except for ”Einstein” and ”Shakespeare”, which have no gramatical gender marking in Spanish. This is most likely the reason for these results and, along with Tests 6 and 7, is further indication of the need of intrinsic bias tests designed specifically designed for Spanish. Save for the outliers, the bias in this test is statistically insignificant.

Target Set 1	Target Set 2	Attribute Set 1	Attribute Set 2
ecuaciones	drama *	niño	ella
álgebra *	poesía *	hermano	hembra
números	literatura *	su	mujer
cálculo	sinfonía *	masculino	hija
geometría *	danza *	hombre	niña
adición	novela *	hijo	su
matemáticas *	arte	él	de ella
	escultura *		hermana

Table 4.4: Lists of target and attribute sets in Spanish for Test 7 of XWEAT. We mark with * the items from the target sets that have female grammatical gender.

Most of the values of the extrinsic bias metrics indicate that the system tends to perform better when dealing with hate speech against women than with hate speech against migrants. As explained previously, Gonen et. al. [18] show that words in

	FastText	word2vec
Test 1	High	High
Test 2	Low	High
Test 6	High	Low
Test 7	Very Low	Very Low
Test 8	Insignificant	Insignificant
Test 9	High	High

Table 4.5: Statistical significance in the bias found on the XWEAT tests. Note that despite this, each test had at least one result with high bias and high significance.

languages that have grammatical gender tend to cluster around that instead of around semantic similarities. We believe that this might be the reason for why our system shows such a disparity between the two groups.

4.3.2 Equality of Opportunity

4.3.2.1 Subtask A - Binary Hate Speech Detection

The equality of opportunity and difference between precision metrics for subtask A had its lowest value (that is, the least bias) when we used both of the unmodified embeddings. However, the range of variations between these results is small, of at most 0.07.

When using FastText, we notice that most of the XWEAT effect sizes have a negative correlation to equality of opportunity of under -0.2. Similarly, there are negative correlations to the difference between precisions, but those have wildly varying numbers, from -0.02 in Test 9 all the way to -0.5 in Test 2. The reason for the negative correlations could be because of the fact that hate messages in social media tend to rely heavily on stereotypes. One way to test this hypothesis would be to label our dataset for different kinds of stereotypes and/or topics and check how our model performs for each of these new labels. The only exception to this trend would be Test 6, which has a positive correlation of over 0.2 for both metrics. This could be an outlier due to the names in the target lists not being common names in Spanish speaking countries, as mentioned before.

In terms of word2vec, the correlations for most of the values were close to 0. This means that our intrinsic and extrinsic metrics interact differently among these two algo-

rithms. The main outliers here were Tests 2 and 6, which had strong negative correlations to equality of opportunity, and Tests 2 and 9, which had had negative correlations to the difference between precisions. On the other hand, Test 7 had a small positive correlation for both metrics, while Test 1 had a strong positive correlation with the difference between precisions. In the end, the results when using word2vec were mostly inconclusive.

4.3.2.2 Subtask B - Agressive Behavior and Target Identification

For this task, our model has to take a tweet, classify whether it is hateful or not, and if it is, to determine whether it is targeted towards a specific individual or not and whether it is agressive or not. We will study how our extrinsic metrics interact with these labels and with the average of them. This last is because the creators of the dataset ask people to report the average F1-score as the performance metric for this subtask. It would also help us get an idea of the general extrinsic biases present across all labels.

When dealing with subtask B, we get a relatively mixed bag of results. As mentioned in section 4.2, the values for equality of opportunity for hate speech detection are very low while the values for the difference between precisions is nearly 0 for all of our experiments. Therefore, we can conclude that there is no significant bias on this label. The label that identifies whether a tweet is targeted or not has bias values similar to the ones found in task A, including the ones coming from the unaltered embeddings. For both labels there is a positive correlation between the intrinsic metrics and both of our extrinsic metrics. The main exceptions to this are Tests 7 and 8 for all metrics in both kinds of embeddings. However, it is important to note that the intrinsic bias for both of these tests was statistically insignificant and that in the case for Test 8, the values just clustered around relatively small areas. Therefore, we can consider that decreasing the intrinsic bias tends to decrease extrinsic bias for these two labels.

We can notice that the aggressiveness label has really mixed results for FastText and generally negative correlations for word2vec. In terms of FastText, we got positive correlations for Tests 1 and 2. The attribute lists for these tests are pleasant and unpleasant, which could be the reason for these results, especially when considering that Test 2 has instruments and weapons as targets. When considering the other XWEAT tests, we find opposite correlations in both of our extrinsic metrics. With Test 6, we get a negative correlation of -0.36 with equality of opportunity and a small positive correlation of 0.17 with the difference of precisions. On the other hand, for Tests 7, 8, and 9 we get strong negative correlations for the difference between pecissions and

small, positive correlations for equality of opportunity. In order to be able to look deeper into the meaning of this, we would have to study the relationship between both of our extrinsic measures, which is outside the scope of this work.

When using word2vec, we can see that most of the correlations are negative for both extrinsic metrics. As with FastText, it could be because the more intrinsic bias present in the embeddings themselves, the more easily our model can pick up aggressiveness in our downstream task. The same caveat about Test 6 being an outlier applies, probably due to the same reason.

When we get to the average extrinsic metrics, we get even more completely different spreads of the correlations. With FastText, we get results similar to those from the aggressiveness label. The main difference is that the positive correlations are stronger and the negative ones are weaker. We can consider that the same analysis for those values apply here.

On the other hand, for word2vec we see positive correlations for all tests with regards to equality of opportunity and negative correlations for the difference between precisions, with the exception of Test 6. As mentioned above, further interpretation of these results would require us to study the relation between our two extrinsic metrics.

In conclusion, both of the embedding algorithms interact in completely different ways with the correlations between our intrinsic and extrinsic metrics. One reason for this could be the fact that FastText uses subword information, which will most likely impact how the bias generated from grammatical gender interacts with topical biases.

For binary hate speech classification we can safely argue that increasing intrinsic bias tends to decrease extrinsic bias by a small amount when using FastText. However, the results are inconclusive when dealing with word2vec.

When doing a more fine-grained classification, we get completely different results depending on the label. This could be an indication on how the more granular hate speech classification can be used to obscure biases and results, especially if we were only to deal with the average extrinsic bias metrics. It is also important to note that different architectures and training approaches might lead to different correlations in these labels. With our values as-is, we can notice that in general hate speech and target detection have direct correlation between their intrinsic and extrinsic biases. However, the opposite is true when dealing with aggressiveness detection when using word2vec, while the results are inconclusive when using FastText.

Chapter 5

Discussion

In this chapter we will talk about how our work slots into the current state of research of bias in NLP and on possible ways to expand upon our work.

One of the places where we could do further exploration is in our intrinsic metric. As we saw from the results from our XWEAT tests, the ones most closely related to gender stereotypes consistently lead to non-statistically relevant results. Zhou et. al. [51] note that the original WEAT tests and the translated XWEAT are ill-equiped to study gender bias in languages with grammatical gender. In order to attempt to solve this, they created a set of tests called MWEAT that are meant to study intrinsic gender bias as it is reflected in the Spanish language in terms of stereotypes instead of grammatical gender. Another thing to note is that the original WEAT test was made considering topical bias as is present in American culture. Therefore, the XWEAT translations that we used do not necessarily reflect the same biases that exist in Spanish-speaking countries. This goes on to highlight that most languages do not have the same NLP resources available that English has. Even worse, some of the resources that are available do not properly take into account cultural and syntactical characteristics of the language. Because of that, it is important to develop new resources that are tailored to the needs and characteristics of other languages.

As for our extrinsic bias metric, we got a mixed bag of negative correlations and inconclusive results, depending of the subtask and the kind of embedding used. One way to expand upon this could be by obtaining mode modified embeddings and thus having more datapoints to take into account. This would be particularly useful to see the correlations with the XWEAT tests that deal with gender bias, as they tended to have low p-values except on experiments made to modify them. The one thing that we noticed was that the relationship between XWEAT and equality of opportunity in

the Spanish language is unreliable. This is a valuable result that indicates that further study should go into studying the relationships between different intrinsic and extrinsic metrics of bias. A possible way to expand upon this would be to use another extrinsic metric, such as pinned area under the curve [13].

Finally, another way to expand our research would be by using other embedding algorithms, including contextual embeddings. This could help us inform on which characteristics of the embedding algorithms are the most important for getting or not noticeable correlations across the board.

Chapter 6

Conclusions

As we saw during 2.1, bias and discrimination have noticeable effects both in machine learning systems and in real life. Moreover, hate speech in social media is a rampant issue that should be addressed.

By studying how these biases interact with different parts of NLP systems and how they affect the fairness of the system, we can make bigger strides in stomping this issue. The results of our work point to an unreliable relation between our intrinsic and extrinsic bias metrics. This is a valuable result as it means that we should not blindly alter intrinsic bias and expect predictable results in the fairness of our system. This is a first step to analyzing these metrics and how much trust is placed upon them.

At the end it is important to remember that machine learning is just a tool. It is our duty to properly understand it to be able to use it for good, instead of causing damage through misuse and ignorance.

Bibliography

[1] Central American Migration Facts | Central American Aid, May 2019. https://www.mercycorps.org/blog/quick-facts-central-american-migration.

[2] Vanessa Barbara. Opinion | Latin America's Radical Feminism Is Spreading. *The New York Times*, January 2020.

[3] Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics.

[4] Nina Bauwelinck, Gilles Jacobs, Véronique Hoste, and Els Lefever. LT3 at SemEval-2019 Task 5: Multilingual Detection of Hate Speech Against Immigrants and Women in Twitter (hatEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 436–440, Minneapolis, Minnesota, USA, 2019. Association for Computational Linguistics.

[5] Su Lin Blodgett, Solon Barocas, Hal Daumé, III, and Hanna Wallach. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. *arXiv e-prints*, 2005:arXiv:2005.14050, May 2020.

[6] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5:135–146, December 2017. Publisher: MIT Press.

[7] 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, pages 4356–4364, Barcelona, Spain, December 2016. Curran Associates Inc.

[8] Tamara Taraciuk Broner. The Venezuelan Exodus, September 2018. https://www.hrw.org/report/2018/09/04/venezuelan-exodus/need-regional-response-unprecedented-migration-crisis (Accessed: 2020-08-21).

[9] Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, April 2017. Publisher: American Association for the Advancement of Science Section: Reports.

[10] Andrew M. Campbell. An increasing risk of family violence during the Covid-19 pandemic: Strengthening community collaborations to save lives. *Forensic Science International: Reports*, 2:100089, December 2020.

[11] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. Racial Bias in Hate Speech and Abusive Language Detection Datasets. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 25–35, Florence, Italy, August 2019. Association for Computational Linguistics.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*, May 2019. arXiv: 1810.04805.

[13] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and Mitigating Unintended Bias in Text Classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, pages 67–73, New Orleans, LA, USA, December 2018. Association for Computing Machinery.

[14] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, ITCS '12, pages 214–226, New York, NY, USA, January 2012. Association for Computing Machinery.

[15] Frank Edwards, Hedwig Lee, and Michael Esposito. Risk of being killed by police use of force in the United States by age, race–ethnicity, and

sex. *Proceedings of the National Academy of Sciences*, 116(34):16793–16798, 2019. Publisher: National Academy of Sciences _eprint: https://www.pnas.org/content/116/34/16793.full.pdf.

[16] European Commission against Racism and Intolerance (ECRI). Hate speech and violence. https://www.coe.int/en/web/european-commission-against-racism-and-intolerance/hate-speech-and-violence (Accessed: 2020-08-21).

[17] Gender Equality Observatory and for Latin America and the Caribbean. Femicide or feminicide - Gender Equality Observatory for Latin America and the Caribbean, February 2016. https://oig.cepal.org/en/indicators/femicide-or-feminicide (Accessed: 2020-08-21).

[18] Hila Gonen, Yova Kementchedjhieva, and Yoav Goldberg. How Does Grammatical Gender Affect Noun Representations in Gender-Marking Languages? In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 463–471, Hong Kong, China, November 2019. Association for Computational Linguistics.

[19] Naiara Galarraga Gortázar. La ruta migratoria a España se convierte en la más letal del mundo. *El País*, May 2018.

[20] Anthony G Greenwald, Debbie E McGhee, and Jordan LK Schwartz. Measuring individual differences in implicit cognition: the implicit association test. *Journal of personality and social psychology*, 74(6):1464, 1998. Publisher: American Psychological Association.

[21] Moritz Hardt, Eric Price, and Nathan Srebro. Equality of opportunity in supervised learning. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 3323–3331, Barcelona, Spain, December 2016. Curran Associates Inc.

[22] Redação Hypeness. 'Não é não': campanha contra assédio no Carnaval atinge 15 estados, January 2020. https://www.hypeness.com.br/2020/01/nao-e-nao-campanha-contra-assedio-no-carnaval-atinge-15-estados/ (Accessed: 2020-08-21).

[23] J. F. I. Root Causes of Migration.

[24] Yoon Kim. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar, October 2014. Association for Computational Linguistics.

[25] Aislinn Laing, Natalia Ramos, Julia Symes Cobb, Marina Lammertyn, and Daina Beth Solomon. Latin American women prepare for record feminist marches. *Reuters*, March 2020.

[26] Anne Lauscher and Goran Glavaš. Are We Consistently Biased? Multidimensional Analysis of Biases in Distributional Word Vectors. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 85–91, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[27] Kalev Leetaru. Is Twitter's Spritzer Stream Really A Nearly Perfect 1% Sample Of Its Firehose?, February 2019. https://www.forbes.com/sites/kalevleetaru/2019/02/27/is-twitters-spritzer-stream-really-a-nearly-perfect-1-sample-of-its-firehose/ (Accessed: 2020-08-21).

[28] Kristian Lum and William Isaac. To predict and serve? *Significance*, 13(5):14–19, 2016. _eprint: https://rss.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1740-9713.2016.00960.x.

[29] Charis McGowan. Chilean anti-rape anthem becomes international feminist phenomenon. *The Guardian*, December 2019.

[30] Bethan McKernan. Challenge accepted: Turkish feminists spell out real meaning of hashtag. *The Guardian*, July 2020.

[31] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv:1301.3781 [cs]*, September 2013. arXiv: 1301.3781.

[32] Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. Semantic Specialization of Distributional Word Vector Spaces using Monolingual and Cross-Lingual Constraints.

Transactions of the Association for Computational Linguistics, 5:309–324, December 2017. Publisher: MIT Press.

[33] Chelsea Mtada. Politics: The Timeline of Events that led to the Black Lives Matter Movement 2020., June 2020. https://guap.co.uk/blog/2020/06/19/politics-the-timeline-of-events-that-led-to-the-black-lives-matter-movement-2020/ (Accessed: 2020-08-20).

[34] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, 2014. Association for Computational Linguistics.

[35] María Martín Peregil, Francisco. "En Argelia ya no nos queda nada". *EL PAÍS*, August 2020. Section: España.

[36] Billy Perrigo. Facebook Says It's Removing More Hate Speech Than Ever Before. But There's a Catch. *Time*, November 2019. https://time.com/5739688/facebook-hate-speech-languages/ (Accessed: 2020-08-21).

[37] Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[38] Laurette Pretorius and Sonja E Bosch. Enabling Computer Interaction in the Indigenous Languages of South Africa: The Central Role of Computational Morphology. *Interactions*, 10:56–63, March 2003.

[39] Natalie Alcoba Charis McGowan in Santiago. #NiUnaMenos five years on: Latin America as deadly as ever for women, say activists. *The Guardian*, June 2020.

[40] Kirk Semple. Overflowing Toilets, Bedbugs and High Heat: Inside Mexico's Migrant Detention Centers. *The New York Times*, August 2019.

[41] Adam Serwer. A Crime by Any Name, July 2019. https://www.theatlantic.com/ideas/archive/2019/07/border-facilities/593239/ (Accessed: 2020-08-21).

[42] Michael R. Smith and Geoffrey P. Alpert. Explaining Police Bias: A Theory of Social Conditioning and Illusory Correlation. *Criminal Justice and Behavior*, 34(10):1262–1283, October 2007. Publisher: SAGE Publications Inc.

[43] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and Policy Considerations for Deep Learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy, July 2019. Association for Computational Linguistics.

[44] UNICEF. Migration, hate speech and media ethics, November 2017. https://blogs.unicef.org/evidence-for-action/migration-hate-speech-and-media-ethics/ (Accessed: 2020-08-20).

[45] UNICEF LACRO. Migration flows in Latin America and the Caribbean Situation Report No. 5, October 2018. https://www.unicef.org/lac/en/reports/migration-flows-latin-america-and-caribbean (Accessed: 2020-08-21).

[46] United Nations Human Rights Office of the High Comissioner. OHCHR | Combating Discrimination against Migrants. https://www.ohchr.org/EN/Issues/Discrimination/Pages/discrimination_migrants.aspx (Accessed: 2020-08-20).

[47] Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. Balanced Datasets Are Not Enough: Estimating and Mitigating Gender Bias in Deep Image Representations. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2018.

[48] Zeerak Waseem. Are You a Racist or Am I Seeing Things? Annotator Influence on Hate Speech Detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas, November 2016. Association for Computational Linguistics.

[49] Ye Zhang and Byron Wallace. A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification. *arXiv:1510.03820 [cs]*, April 2016. arXiv: 1510.03820.

[50] Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 15–20, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[51] Pei Zhou, Weijia Shi, Jieyu Zhao, Kuan-Hao Huang, Muhao Chen, Ryan Cotterell, and Kai-Wei Chang. Examining Gender Bias in Languages with Grammatical Gender. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5276–5284, Hong Kong, China, November 2019. Association for Computational Linguistics.

[52] Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50, Valletta, Malta, May 2010. ELRA.

Appendix A

Appendix A: Results

4.2. Here we are including the results from different parts of our pipeline. On the main body of the dissertation, in section 4.2, we included the most relevant values that we were looking at. The rest of them are in this section so as to not to overwhelm the reader with the sheer amount of data.

A.1 XWEAT Tests

		Test 1			Test 2			Test 6			Test 7			Test 8			Test 9		
		WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value
Unmodified Embedding		0.74	1.41	0.00	0.18	0.44	0.16	0.30	1.01	0.03	0.16	0.57	0.18	-0.02	-0.06	0.54	0.24	1.12	0.03
Test 1	More Bias	14.74	2.00	0.00	0.65	0.56	0.10	0.33	1.03	0.03	0.16	0.59	0.17	-0.02	-0.06	0.54	0.27	1.35	0.01
	Less Bias	-12.20	-2.00	1.00	-0.31	-0.32	0.75	0.31	1.03	0.03	0.15	0.55	0.19	-0.02	-0.06	0.54	0.30	1.60	0.00
Test 2	More Bias	2.40	1.76	0.00	12.09	1.95	0.00	0.35	1.12	0.02	0.21	0.82	0.09	-0.02	-0.06	0.54	0.28	1.48	0.00
	Less Bias	0.43	0.58	0.09	-11.63	-2.00	1.00	0.41	1.41	0.00	0.16	0.59	0.17	-0.02	-0.06	0.54	0.08	0.46	0.23
Test 6	More Bias	0.74	1.38	0.00	0.20	0.46	0.15	0.31	1.03	0.03	0.13	0.50	0.22	-0.02	-0.06	0.54	0.44	1.47	0.00
	Less Bias	0.70	1.34	0.00	0.09	0.21	0.32	0.35	1.18	0.02	0.16	0.58	0.17	-0.02	-0.06	0.54	0.17	1.17	0.02
Test 7	More Bias	0.70	1.34	0.00	0.09	0.21	0.32	0.30	1.07	0.02	1.97	1.93	0.00	0.55	1.24	0.00	0.31	1.69	0.00
	Less Bias	0.74	1.41	0.00	0.14	0.31	0.24	0.30	1.16	0.01	-1.70	-1.93	1.00	-0.68	-1.53	1.00	0.30	1.21	0.02
Test 8	More Bias	0.73	1.41	0.00	0.16	0.37	0.21	0.28	1.03	0.03	0.57	1.39	0.01	1.42	1.79	0.00	0.29	1.17	0.02
	Less Bias	0.79	1.49	0.00	0.28	0.67	0.06	0.40	1.45	0.00	-0.35	-1.25	0.99	-1.55	-1.79	1.00	0.30	1.21	0.02
Test 9	More Bias	0.69	1.34	0.00	0.20	0.46	0.15	0.38	1.16	0.02	0.25	0.93	0.07	-0.02	-0.06	0.54	3.09	1.96	0.00
	Less Bias	0.73	1.38	0.00	0.23	0.53	0.12	0.39	1.24	0.01	0.25	0.93	0.07	-0.02	-0.06	0.54	-2.67	-1.95	1.00

Table A.1: Results from the XWEAT tests (columns) from the different embeddings that we used (rows) when using FastText.

		Test 1			Test 2			Test 6			Test 7			Test 8			Test 9		
		WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value
Unmodified Embedding		0.38	0.89	0.01	0.47	0.81	0.03	0.33	0.73	0.10	0.16	0.70	0.14	0.06	0.26	0.32	0.33	1.13	0.03
Test 1	More Bias	16.39	2.00	0.00	0.82	0.53	0.11	0.32	0.78	0.08	0.15	0.72	0.13	0.06	0.26	0.32	0.60	1.58	0.00
	Less Bias	-15.82	-2.00	1.00	0.57	0.56	0.11	0.36	0.81	0.07	0.16	0.70	0.14	0.06	0.26	0.32	0.62	1.59	0.00
Test 2	More Bias	1.38	1.07	0.00	14.22	1.95	0.00	0.35	0.84	0.07	0.14	0.67	0.14	0.06	0.26	0.32	0.56	1.18	0.02
	Less Bias	-0.12	-0.16	0.65	-13.63	-2.00	1.00	0.30	0.69	0.11	0.11	0.55	0.19	0.06	0.26	0.32	0.48	1.12	0.03
Test 6	More Bias	0.38	0.90	0.01	0.49	0.84	0.03	0.27	0.66	0.12	0.22	0.88	0.08	0.06	0.26	0.32	0.53	1.61	0.00
	Less Bias	0.49	1.10	0.00	0.44	0.75	0.04	0.36	0.91	0.05	0.22	0.88	0.08	0.06	0.26	0.32	0.52	1.28	0.02
Test 7	More Bias	0.40	0.92	0.01	0.57	0.99	0.01	0.26	0.62	0.14	2.57	1.94	0.00	0.71	1.53	0.00	0.33	1.13	0.03
	Less Bias	0.49	1.09	0.00	0.41	0.77	0.04	0.28	0.66	0.12	-2.09	-1.92	1.00	-0.76	-1.60	1.00	0.26	0.91	0.06
Test 8	More Bias	0.53	1.24	0.00	0.53	0.91	0.02	0.32	0.76	0.09	0.75	1.72	0.00	1.97	1.83	0.00	0.71	1.61	0.00
	Less Bias	0.45	1.04	0.01	0.52	0.95	0.01	0.31	0.71	0.10	-0.45	-1.21	0.97	-1.89	-1.82	1.00	0.82	1.39	0.01
Test 9	More Bias	0.36	0.88	0.02	0.50	0.86	0.02	0.28	0.66	0.12	0.18	0.74	0.12	0.06	0.26	0.32	3.59	1.94	0.00
	Less Bias	0.34	0.87	0.02	0.42	0.73	0.05	0.25	0.62	0.13	0.17	0.75	0.12	0.06	0.26	0.32	-2.88	-1.97	1.00

Table A.2: Results from the XWEAT tests (columns) from the different embeddings that we used (rows) when using word2vec.

A.2 HatEval Subtask A - Hate Speech Detection

		All Data				Group 1 Hate Speech Against Women				Group 1 Hate Speech Against Women			
		WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value
Unmodified Embedding		0.76	0.76	0.76	0.76	0.86	0.86	0.86	0.86	0.67	0.66	0.66	0.66
Test 1	More Bias	0.75	0.74	0.74	0.74	0.86	0.86	0.86	0.86	0.67	0.62	0.62	0.62
	Less Bias	0.75	0.74	0.74	0.74	0.86	0.86	0.86	0.86	0.66	0.62	0.62	0.62
Test 2	More Bias	0.75	0.74	0.75	0.74	0.86	0.85	0.85	0.85	0.66	0.63	0.64	0.63
	Less Bias	0.75	0.75	0.75	0.75	0.86	0.86	0.86	0.86	0.66	0.64	0.64	0.64
Test 6	More Bias	0.75	0.74	0.74	0.74	0.86	0.86	0.86	0.86	0.66	0.62	0.62	0.62
	Less Bias	0.75	0.74	0.74	0.74	0.87	0.87	0.86	0.87	0.66	0.62	0.62	0.62
Test 7	More Bias	0.74	0.73	0.74	0.73	0.85	0.85	0.84	0.85	0.65	0.62	0.63	0.62
	Less Bias	0.75	0.74	0.74	0.74	0.86	0.86	0.86	0.86	0.65	0.62	0.63	0.62
Test 8	More Bias	0.75	0.75	0.75	0.75	0.86	0.86	0.86	0.86	0.66	0.64	0.64	0.64
	Less Bias	0.74	0.74	0.74	0.74	0.85	0.85	0.85	0.85	0.65	0.63	0.63	0.63
Test 9	More Bias	0.75	0.75	0.75	0.75	0.86	0.85	0.85	0.85	0.67	0.65	0.65	0.65
	Less Bias	0.76	0.73	0.73	0.73	0.86	0.86	0.86	0.86	0.67	0.60	0.59	0.60

Table A.3: Results from the HatEval subtask A from the different embeddings when using FastText.

		All Data				Group 1 Hate Speech Against Women				Group 1 Hate Speech Against Women			
		WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value	WEAT Statistic	Effect Size	p-Value
Unmodified Embedding		0.76	0.75	0.75	0.75	0.85	0.85	0.85	0.85	0.67	0.66	0.66	0.66
Test 1	More Bias	0.74	0.74	0.74	0.74	0.86	0.86	0.85	0.86	0.64	0.63	0.63	0.63
	Less Bias	0.76	0.76	0.76	0.76	0.86	0.86	0.86	0.86	0.67	0.66	0.66	0.66
Test 2	More Bias	0.75	0.75	0.75	0.75	0.85	0.85	0.85	0.85	0.66	0.65	0.66	0.65
	Less Bias	0.75	0.74	0.75	0.74	0.86	0.86	0.86	0.86	0.66	0.63	0.63	0.63
Test 6	More Bias	0.75	0.74	0.74	0.74	0.85	0.85	0.84	0.85	0.67	0.63	0.63	0.63
	Less Bias	0.75	0.75	0.75	0.75	0.86	0.86	0.86	0.86	0.65	0.64	0.64	0.64
Test 7	More Bias	0.75	0.74	0.75	0.74	0.86	0.86	0.86	0.86	0.65	0.63	0.63	0.63
	Less Bias	0.75	0.74	0.75	0.74	0.86	0.85	0.85	0.85	0.65	0.64	0.64	0.64
Test 8	More Bias	0.76	0.76	0.76	0.76	0.86	0.86	0.86	0.86	0.66	0.66	0.66	0.66
	Less Bias	0.75	0.75	0.75	0.75	0.86	0.86	0.86	0.86	0.66	0.64	0.64	0.64
Test 9	More Bias	0.74	0.72	0.72	0.72	0.85	0.85	0.84	0.85	0.67	0.59	0.58	0.59
	Less Bias	0.76	0.75	0.75	0.75	0.86	0.86	0.86	0.86	0.66	0.63	0.64	0.63

Table A.4: Results from the HatEval subtask A from the different embeddings when using word2vec.

A.3 HatEval Subtask B - Agressive Behavior and Target Identification

		Hate Speech				Individual Target				Aggressiveness				General	
		Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	EMR	F1-Score
Unmodified Embedding		1	0.762	0.861	76.414	0.837	0.821	0.821	76.414	0.724	0.695	0.699	76.414	0.545	0.794
Test 1	More Bias	1	0.795	0.883	76.667	0.822	0.825	0.818	76.667	0.718	0.668	0.659	76.667	0.544	0.787
	Less Bias	1	0.789	0.879	77.424	0.861	0.852	0.848	77.424	0.724	0.68	0.691	77.424	0.559	0.806
Test 2	More Bias	1	0.728	0.839	73.535	0.825	0.815	0.812	73.535	0.738	0.662	0.676	73.535	0.509	0.776
	Less Bias	1	0.752	0.855	74.697	0.827	0.813	0.811	74.697	0.709	0.661	0.664	74.697	0.518	0.777
Test 6	More Bias	1	0.787	0.878	77.677	0.829	0.816	0.814	77.677	0.746	0.712	0.714	77.677	0.567	0.802
	Less Bias	1	0.748	0.852	74.141	0.826	0.828	0.818	74.141	0.727	0.641	0.646	74.141	0.509	0.772
Test 7	More Bias	1	0.797	0.884	77.525	0.82	0.826	0.817	77.525	0.73	0.689	0.69	77.525	0.559	0.797
	Less Bias	1	0.716	0.83	72.929	0.822	0.796	0.799	72.929	0.72	0.664	0.677	72.929	0.509	0.768
Test 8	More Bias	1	0.833	0.907	80.051	0.838	0.835	0.829	80.051	0.687	0.711	0.613	80.051	0.571	0.783
	Less Bias	1	0.797	0.884	76.919	0.824	0.806	0.806	76.919	0.707	0.685	0.679	76.919	0.552	0.79
Test 9	More Bias	1	0.746	0.849	74.95	0.824	0.808	0.805	74.95	0.739	0.685	0.696	74.95	0.529	0.784
	Less Bias	1	0.739	0.846	74.293	0.834	0.821	0.815	74.293	0.737	0.667	0.676	74.293	0.517	0.779

Table A.5: Results from the HatEval subtask B from the different embeddings when using FastText.

		Hate Speech				Individual Target				Aggressiveness				General	
		Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	EMR	F1-Score
Unmodified Embedding		1	0.789	0.879	85.218	0.938	0.951	0.937	85.218	0.901	0.794	0.834	85.218	0.747	0.883
Test 1	More Bias	1	0.805	0.89	85.516	0.959	0.958	0.958	85.516	0.898	0.802	0.84	85.516	0.747	0.896
	Less Bias	1	0.768	0.865	83.234	0.947	0.958	0.948	83.234	0.892	0.755	0.806	83.234	0.714	0.873
Test 2	More Bias	1	0.701	0.818	79.96	0.941	0.956	0.946	79.96	0.897	0.708	0.774	79.96	0.673	0.846
	Less Bias	1	0.737	0.843	82.242	0.932	0.948	0.937	82.242	0.896	0.74	0.795	82.242	0.702	0.858
Test 6	More Bias	1	0.794	0.882	85.218	0.93	0.945	0.936	85.218	0.901	0.794	0.834	85.218	0.741	0.884
	Less Bias	1	0.719	0.833	79.861	0.941	0.951	0.939	79.861	0.897	0.724	0.787	79.861	0.67	0.853
Test 7	More Bias	1	0.792	0.882	84.821	0.944	0.953	0.942	84.821	0.901	0.794	0.836	84.821	0.741	0.886
	Less Bias	1	0.732	0.841	81.448	0.938	0.953	0.944	81.448	0.899	0.734	0.794	81.448	0.688	0.86
Test 8	More Bias	1	0.807	0.89	89.98	0.936	0.948	0.94	89.98	0.861	0.927	0.892	89.98	0.759	0.908
	Less Bias	1	0.818	0.898	86.409	0.939	0.953	0.944	86.409	0.896	0.81	0.844	86.409	0.759	0.895
Test 9	More Bias	1	0.776	0.869	83.631	0.932	0.948	0.937	83.631	0.897	0.768	0.815	83.631	0.714	0.874
	Less Bias	1	0.716	0.83	80.754	0.932	0.948	0.937	80.754	0.898	0.721	0.783	80.754	0.679	0.85

Table A.6: Results from the HatEval subtask B from the different embeddings when using FastText. We are only testing on hate speech against women in this table.

		Hate Speech				Individual Target				Aggressiveness				General	
		Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	EMR	F1-Score
Unmodified Embedding		1	0.758	0.857	67.078	0.722	0.716	0.708	67.078	0.549	0.576	0.543	67.078	0.327	0.703
Test 1	More Bias	1	0.805	0.888	68.416	0.714	0.721	0.709	68.416	0.549	0.557	0.509	68.416	0.346	0.702
	Less Bias	1	0.818	0.896	71.193	0.809	0.789	0.792	71.193	0.56	0.586	0.555	71.193	0.395	0.748
Test 2	More Bias	1	0.763	0.859	66.975	0.688	0.688	0.677	66.975	0.578	0.596	0.566	66.975	0.343	0.701
	Less Bias	1	0.773	0.867	67.284	0.713	0.701	0.693	67.284	0.568	0.581	0.54	67.284	0.333	0.7
Test 6	More Bias	1	0.792	0.879	69.547	0.719	0.714	0.704	69.547	0.589	0.607	0.571	69.547	0.38	0.718
	Less Bias	1	0.794	0.881	68.724	0.724	0.724	0.711	68.724	0.565	0.573	0.533	68.724	0.355	0.708
Test 7	More Bias	1	0.823	0.899	70.885	0.707	0.724	0.707	70.885	0.588	0.599	0.561	70.885	0.383	0.722
	Less Bias	1	0.716	0.826	63.889	0.708	0.674	0.671	63.889	0.556	0.578	0.548	63.889	0.321	0.682
Test 8	More Bias	1	0.859	0.922	69.856	0.719	0.74	0.721	69.856	0.598	0.521	0.376	69.856	0.373	0.673
	Less Bias	1	0.779	0.869	66.049	0.701	0.688	0.681	66.049	0.542	0.557	0.518	66.049	0.324	0.689
Test 9	More Bias	1	0.737	0.84	66.564	0.717	0.695	0.689	66.564	0.584	0.604	0.574	66.564	0.352	0.701
	Less Bias	1	0.755	0.855	66.872	0.72	0.708	0.698	66.872	0.56	0.581	0.546	66.872	0.336	0.7

Table A.7: Results from the HatEval subtask B from the different embeddings when using FastText. We are only testing on hate speech against migrants in this table.

		Hate Speech				Individual Target				Aggressiveness				General	
		Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	EMR	F1-Score
Unmodified Embedding		1	0.724	0.833	73.384	0.838	0.816	0.814	73.384	0.717	0.661	0.672	73.384	0.508	0.773
Test 1	More Bias	1	0.701	0.82	73.182	0.826	0.807	0.803	73.182	0.736	0.666	0.686	73.182	0.517	0.77
	Less Bias	1	0.711	0.827	72.273	0.839	0.824	0.819	72.273	0.697	0.629	0.647	72.273	0.488	0.764
Test 2	More Bias	1	0.809	0.892	78.232	0.84	0.839	0.832	78.232	0.731	0.689	0.693	78.232	0.559	0.806
	Less Bias	1	0.781	0.873	75.96	0.817	0.826	0.815	75.96	0.72	0.666	0.677	75.96	0.535	0.788
Test 6	More Bias	1	0.794	0.881	77.576	0.857	0.849	0.842	77.576	0.725	0.676	0.675	77.576	0.556	0.799
	Less Bias	1	0.72	0.831	73.384	0.822	0.803	0.801	73.384	0.738	0.678	0.69	73.384	0.508	0.774
Test 7	More Bias	1	0.762	0.861	76.01	0.831	0.832	0.824	76.01	0.736	0.684	0.695	76.01	0.541	0.793
	Less Bias	1	0.787	0.877	77.071	0.83	0.829	0.822	77.071	0.738	0.692	0.694	77.071	0.547	0.797
Test 8	More Bias	1	0.697	0.814	71.465	0.817	0.786	0.789	71.465	0.712	0.652	0.672	71.465	0.483	0.758
	Less Bias	1	0.738	0.844	74.394	0.844	0.828	0.821	74.394	0.741	0.67	0.685	74.394	0.52	0.783
Test 9	More Bias	1	0.755	0.855	75.606	0.833	0.819	0.815	75.606	0.75	0.692	0.7	75.606	0.541	0.79
	Less Bias	1	0.709	0.823	72.475	0.835	0.829	0.822	72.475	0.708	0.638	0.655	72.475	0.489	0.767

Table A.8: Results from the HatEval subtask B from the different embeddings when using word2vec.

		Hate Speech				Individual Target				Aggressiveness				General	
		Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	EMR	F1-Score
Unmodified Embedding		1	0.75	0.851	82.639	0.928	0.948	0.935	82.639	0.895	0.753	0.804	82.639	0.705	0.863
Test 1	More Bias	1	0.724	0.835	80.952	0.94	0.953	0.939	80.952	0.886	0.714	0.777	80.952	0.682	0.85
	Less Bias	1	0.716	0.83	80.06	0.928	0.948	0.935	80.06	0.886	0.703	0.77	80.06	0.664	0.845
Test 2	More Bias	1	0.771	0.866	84.821	0.929	0.945	0.935	84.821	0.902	0.776	0.822	84.821	0.735	0.874
	Less Bias	1	0.758	0.857	83.135	0.928	0.948	0.935	83.135	0.89	0.742	0.797	83.135	0.705	0.863
Test 6	More Bias	1	0.792	0.88	84.722	0.949	0.958	0.946	84.722	0.89	0.771	0.816	84.722	0.732	0.881
	Less Bias	1	0.734	0.841	81.746	0.931	0.945	0.936	81.746	0.891	0.732	0.789	81.746	0.69	0.855
Test 7	More Bias	1	0.755	0.854	82.937	0.93	0.943	0.934	82.937	0.886	0.745	0.797	82.937	0.708	0.862
	Less Bias	1	0.789	0.878	85.218	0.927	0.945	0.934	85.218	0.897	0.784	0.825	85.218	0.738	0.879
Test 8	More Bias	1	0.742	0.845	81.746	0.934	0.945	0.938	81.746	0.886	0.724	0.783	81.746	0.682	0.855
	Less Bias	1	0.737	0.843	81.647	0.931	0.951	0.937	81.647	0.892	0.721	0.784	81.647	0.682	0.855
Test 9	More Bias	1	0.768	0.863	83.333	0.934	0.951	0.94	83.333	0.884	0.75	0.799	83.333	0.711	0.867
	Less Bias	1	0.703	0.819	79.464	0.928	0.945	0.935	79.464	0.891	0.703	0.768	79.464	0.664	0.841

Table A.9: Results from the HatEval subtask B from the different embeddings when using word2vec. We are only testing on hate speech against women in this table.

		Hate Speech				Individual Target				Aggressiveness				General	
		Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	Precision	Recall	F1-Score	Accuracy	EMR	F1-Score
Unmodified Embedding		1	0.711	0.821	64.403	0.728	0.708	0.703	64.403	0.534	0.562	0.533	64.403	0.312	0.686
Test 1	More Bias	1	0.706	0.818	65.638	0.726	0.701	0.691	65.638	0.579	0.607	0.578	65.638	0.352	0.696
	Less Bias	1	0.719	0.829	65.021	0.737	0.721	0.714	65.021	0.53	0.557	0.528	65.021	0.312	0.69
Test 2	More Bias	1	0.828	0.903	71.399	0.724	0.745	0.727	71.399	0.568	0.586	0.551	71.399	0.377	0.727
	Less Bias	1	0.792	0.878	69.239	0.719	0.732	0.716	69.239	0.647	0.62	0.615	69.239	0.367	0.736
Test 6	More Bias	1	0.794	0.881	69.547	0.756	0.758	0.744	69.547	0.54	0.56	0.523	69.547	0.358	0.716
	Less Bias	1	0.701	0.816	64.3	0.723	0.69	0.686	64.3	0.56	0.589	0.56	64.3	0.312	0.687
Test 7	More Bias	1	0.781	0.871	69.856	0.728	0.74	0.724	69.856	0.665	0.638	0.631	69.856	0.383	0.742
	Less Bias	1	0.779	0.869	68.416	0.723	0.737	0.721	68.416	0.549	0.568	0.532	68.416	0.34	0.708
Test 8	More Bias	1	0.669	0.792	61.008	0.708	0.661	0.661	61.008	0.626	0.604	0.602	61.008	0.281	0.685
	Less Bias	1	0.737	0.841	66.564	0.734	0.716	0.705	66.564	0.655	0.622	0.617	66.564	0.349	0.721
Test 9	More Bias	1	0.737	0.842	67.181	0.717	0.706	0.698	67.181	0.579	0.609	0.581	67.181	0.358	0.707
	Less Bias	1	0.724	0.831	66.255	0.734	0.732	0.721	66.255	0.539	0.573	0.546	66.255	0.324	0.699

Table A.10: Results from the HatEval subtask B from the different embeddings when using word2vec. We are only testing on hate speech against migrants in this table.

A.4 Equality of Opportunity

		Task A		Task B							
				Hate Speech		Individual Target		Aggressiveness		Average	
		Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference
Unmodified Embedding		0.20	0.18	0.03	0.00	0.24	0.22	0.22	0.35	0.16	0.19
Test 1	More Bias	0.23	0.19	0.00	0.00	0.24	0.25	0.25	0.35	0.16	0.20
	Less Bias	0.24	0.19	-0.05	0.00	0.17	0.14	0.17	0.33	0.10	0.16
Test 2	More Bias	0.22	0.19	-0.06	0.00	0.27	0.25	0.11	0.32	0.11	0.19
	Less Bias	0.23	0.20	-0.04	0.00	0.25	0.22	0.16	0.33	0.12	0.18
Test 6	More Bias	0.24	0.19	0.00	0.00	0.23	0.21	0.19	0.31	0.14	0.17
	Less Bias	0.25	0.20	-0.08	0.00	0.23	0.22	0.15	0.33	0.10	0.18
Test 7	More Bias	0.22	0.20	-0.03	0.00	0.23	0.24	0.20	0.31	0.13	0.18
	Less Bias	0.24	0.22	0.02	0.00	0.28	0.23	0.16	0.34	0.15	0.19
Test 8	More Bias	0.22	0.21	-0.05	0.00	0.21	0.22	0.41	0.26	0.19	0.16
	Less Bias	0.22	0.20	0.04	0.00	0.27	0.24	0.25	0.35	0.19	0.20
Test 9	More Bias	0.21	0.19	0.04	0.00	0.25	0.22	0.16	0.31	0.15	0.18
	Less Bias	0.27	0.19	-0.04	0.00	0.24	0.21	0.14	0.34	0.11	0.18

Table A.11: Results from our extrinsic metrics when using FastText.

A.5 Equality of Opportunity

		Task A		Task B							
				Hate Speech		Individual Target		Aggressiveness		Average	
		Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference	Equality of Opportunity	Precisions Difference
Unmodified Embedding		0.19	0.18	0.04	0.00	0.24	0.20	0.19	0.36	0.16	0.19
Test 1	More Bias	0.23	0.22	0.02	0.00	0.25	0.21	0.11	0.31	0.13	0.17
	Less Bias	0.20	0.19	0.00	0.00	0.23	0.19	0.15	0.36	0.12	0.18
Test 2	More Bias	0.20	0.19	-0.06	0.00	0.20	0.21	0.19	0.33	0.11	0.18
	Less Bias	0.24	0.20	-0.03	0.00	0.22	0.21	0.12	0.24	0.10	0.15
Test 6	More Bias	0.22	0.18	0.00	0.00	0.20	0.19	0.21	0.35	0.14	0.18
	Less Bias	0.22	0.21	0.03	0.00	0.26	0.21	0.14	0.33	0.14	0.18
Test 7	More Bias	0.24	0.21	-0.03	0.00	0.20	0.20	0.11	0.22	0.09	0.14
	Less Bias	0.22	0.21	0.01	0.00	0.21	0.20	0.22	0.35	0.14	0.18
Test 8	More Bias	0.20	0.20	0.07	0.00	0.28	0.23	0.12	0.26	0.16	0.16
	Less Bias	0.22	0.19	0.00	0.00	0.24	0.20	0.10	0.24	0.11	0.14
Test 9	More Bias	0.26	0.18	0.03	0.00	0.25	0.22	0.14	0.31	0.14	0.17
	Less Bias	0.23	0.20	-0.02	0.00	0.21	0.19	0.13	0.35	0.11	0.18

Table A.12: Results from our extrinsic metrics when using word2vec.

A.6 Code Used for this Project

In this appendix, we will talk about the code that we used and where it came from.

- Preprocessing - most of the code to filter the Twitter data was original, the only exception would be the tweet preprocessing script, which was created by my coleague Mugdha Pandya[1].
- Embeddings - the code to generate the word embeddings was original, though based on the gensim [52] tutorial for doing so[2].
- XWEAT - this code is largely the same as the original authors'[3] [26]. Some of the hardcoded flags were changed to allow the model to import gensim embeddings.
- Attract-Repel - this is the code from the original paper[4] [32], changed by my coleague Rebecca Marchant[5] to work on Python 3.
- CNN - we used the CNN code by github user Shawn1993[6]. Some parts of the code were heavily modified to be able to use our own dataloader and being able

[1] $https://github.com/seraphinatarrant/embedding_bias/blob/Mugdha/data/scripts/preprocess.ipynb$
[2] $https://radimrehurek.com/gensim/auto_examples/tutorials/run_fasttext.html$
[3] $https://github.com/anlausch/XWEAT$
[4] $https://github.com/nmrksic/attract-repel$
[5] $https://github.com/seraphinatarrant/embedding_bias/tree/Rebecca/attract-repel$
[6] $https://github.com/Shawn1993/cnn-text-classification-pytorch$

to use our pre-trained embeddings. For also modified the evaluation function to report more variables.

- CNN2 - this is essentially the same code as the previous one, but with an even more heavily modified evaluation function. It allows us to train three different CNNs separately and then uses them to cascade classify.