Chloe Zeller

Machine Learning

**Project 4: Independent Project: Using Word Embedding to Measure Objectivity**

**Introduction**

On the very first day of machine learning, we were asked to read a WIRED article called “Machines Taught by Photos Learn a Sexist View of Women” by Tom Simonite (Simonite 2017). I was quite taken with the article because it brought up so many important problems in machine learning and artificial intelligence (AI) overall. Machine learning has capitalized and made great strides by taking advantage of training models using very large sets of data. What we do not consider often enough, however, are the repercussions of teaching an AI based on data that our inherently flawed society produces. Datta et al. found that Google’s Ad preferences manager automatically prioritizes executive level positions to male users over female users, when both users only differed on gender (Datta, Tschantz, and Datta 2015). This is thought to be partly because Google uses word embedding vectors to train it’s Advertising, and the word “executive” is more associated with (i.e. has shorter distance to) “male” than “female.” The same biases that humans inherently have (e.g. racism, sexism, classism, ageism, to name a few), will be inherited by the AI, much as a child would learn from a parent.

**Literature Review and Background**

Over the course of the semester, I have been investigating word embedding, and the growing body of research on using word embedding to detect inherent bias in AIs. First, we need to understand how implicit bias can be quantified in AIs trained using machine learning. The most comprehensive study showing the inheritance of implicit bias is *Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings* by Bolukbasi et al. (2016). In this paper, researchers trained an AI using a large set of Google News articles, using word embedding, and then tested to see if the AI had learned sexist associations. This project relied heavily on the Bolukbasi et al. paper as a source for replication. Specifically, I used their algorithm for measuring direct bias.

Word embedding, broadly, involves taking a single word or phrase, and converting it to a vector. Words that are more semantically related have shorter vector distance to each other than less similar words, making it very easy to compare relationships between words. Once the AI had been trained, Bolukbasi et al. used properties of the word embedding to measure the AI’s gender bias, and they found extreme prevalence of sexism in the AI (e.g., a large margin between the relationships between *man* and *woman*, with gender-neutral occupations like *nurse* or *computer programmer*). Further explanation of how each of the two embeddings I used will come in the **Training Word Embeddings** section.

This project focused on assessing gender bias in two different datasets, Tweets and Google News articles, to see if they differ significantly. This project took the framework and methods used in the Bolukbasi study, and compared measured bias in two models. Specifically, we will be looking at bias in the context of gender neutral occupational titles (e.g. *nurse, doctor, plumber, teacher,* etc.), and adjectives (e.g. *kind, compassionate, violent, driven,* etc.). Once I evaluate the level of gender bias on the basis of occupation in each dataset, I can see whether or not one dataset resulted in a more biased AI. For example, if one dataset presents with a higher Direct Bias measurement than the other, then I can make conclusions about relative bias of each model to the other.

**Training Word Embeddings**

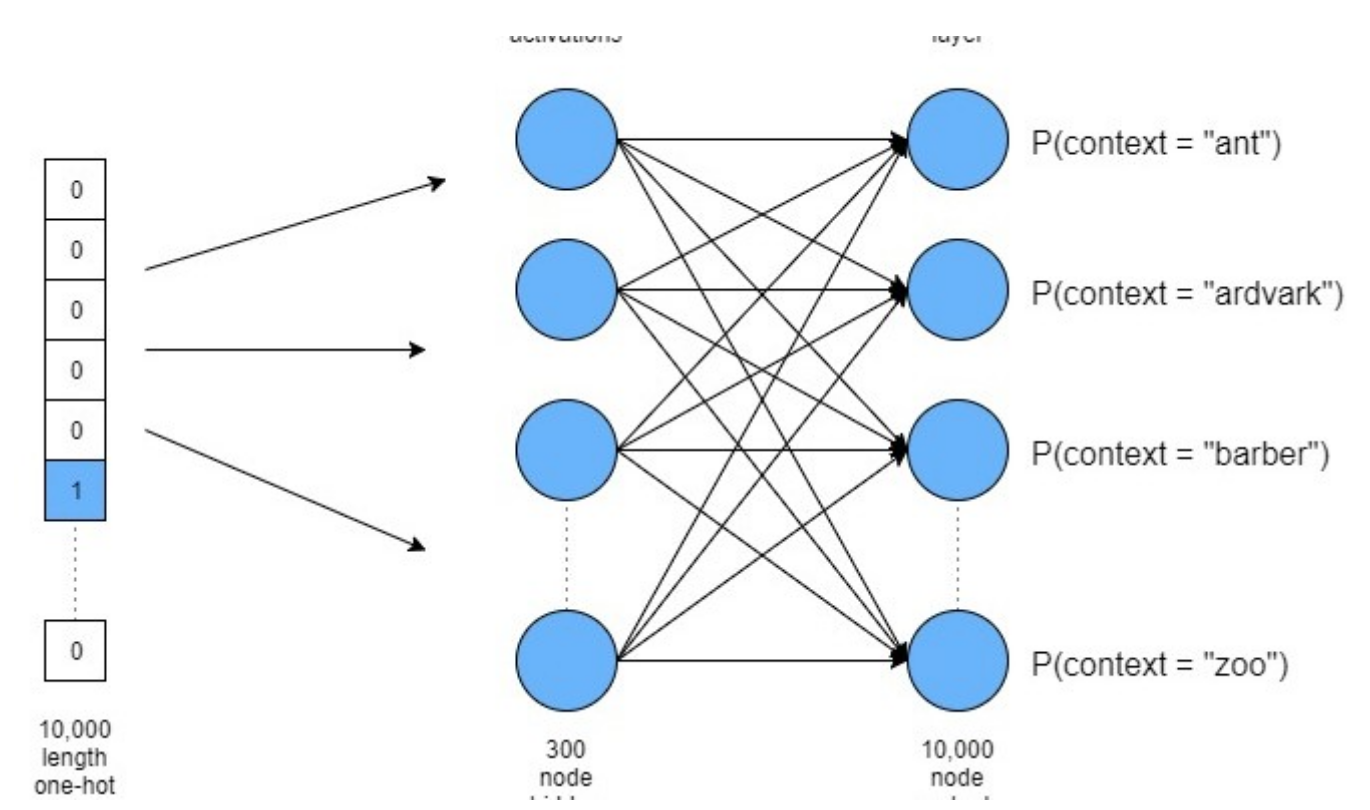
*Word2Vec*

The first model I used was a popular pre-trained word embedding model called *Word2Vec*. Word2Vec is a word embedding that was developed by Tomas Mikolov at Google. The particular Word2Vec pre-trained model I chose was one that had been trained using text from Google News articles (Tomas Mikolov 2013). The model was trained on around 100 billion words, with about 3 million unique words and phrases. Each word and phrase is represented with a 300-dimensional vector. Word2Vec is a pre-trained model, but it’s important to consider the way in which it was trained, especially its limitations when we conduct analysis on its learned bias.

Fundamentally, word embedding relies on word co-occurrence. That is, the number of times that a word appears in the presence of word . And further, we actually want the conditional probability: given word , what is the probability that we will have word in the same sentence, and vice versa (). Ideally, for every word pair we have, we want , however, for even small corpora of text, this would require copious amounts of memory. Alternatively, we compute the pointwise mutual information in lieu of the conditional probability, such that:

The PMI is how much more likely you are to see the pair a and b together than if it were simply at random. Even more simply, the PMI of two vectors is approximately their dot product:

In order to actually process a corpus of text and transform it into these probabilistic representations. We need to implement a softmax neural network.

The inputs into the neural network are called word representation vectors, or “one hot vectors.” If you compute the one-hot vector for the word “cat,” for example, then you could take all sentences in a corpus, and make a vector with dimension equal to the number of sentences. If the sentence contains the word “cat,” then the vector has a 1 at that index, otherwise 0. For Word2Vec, the one hot vectors are 10,000 dimensional because there are 10,000 unique vocabulary words. The one hot vectors are the inputs to the neural network, which is followed by a 300-node hidden layer, have 300 nodes allows dimensionality reduction so that the one hot vectors that were 10,000 dimensional will be 300 dimensional when they are finished running through the network. In the hidden layer, the input vectors are multiplied by activations. Those activations are actually in the form of a 10,000x300 weight matrix where each row is a vector corresponding to a word in the vocabulary. The matrix operations between the 10,000 dimensional one-hot vectors and this matrix allow for dimensionality reduction.

After the linear activation layer, there is a fully connected softmax output layer. In total, the network only has three layers, so it’s quite shallow. Once we’ve finished running the network, the resulting weight matrix is actually the reference we use for subsequent operations. **Figure 1** shows what this neural network’s structure looks like, generally.

Figure : Softmax neural network for a word embedding like Word2Vec (Thomas 2017)

*GloVe*

The second model I used came from the set of pre-trained Global Representation Vectors, or simply, GloVe (Pennington, Socher, and Manning 2014). GloVe, like Word2Vec, has a number of pre-trained models to choose from. I used their model trained on Tweets because I was curious about the comparison or bias between a model trained on tweets and one trained on Google News articles.

GloVe is trained in a manner similar to Word2Vec. They also define a matrix representing the probabilities of word co-occurrence between pairs of words, and then use the log of the probabilities to minimize the matrix space. What is different is that GloVe takes into account that not all co-occurrences are equally relevant. In Word2Vec and other word embedding algorithms, all word co-occurrences are weighted equally, even though it’s very rare that this is the case. Researchers proposed a new “weighted least squares regression model” that is meant to weight the co-occurrences more equitably:

This least-squares regression model replaces the softmax weighting seen in the final layer of the Word2Vec neural network, but other than this change, the models were trained in much the same way.

In their paper, the researchers behind GloVe actually address the Word2Vec model, and how the models differ. While the models differ on how they set their parameters, they do say that “for the same corpus, vocabulary, window size, and training time, GloVe consistently outperforms word2vec. It achieves better results faster, and also obtains the best results irrespective of speed” (Pennington, Socher, and Manning 2014). On the GloVe website, they provide a list of pre-trained models that are trained on a variety of different text corpora. For the purposes of my project, I chose the model trained on Tweets, which contained 2 billion tweets, with 1.2 million unique vocabulary words. I chose the version with the largest dimension vectors because I wanted the strongest vector representation, the vectors were 200 dimensional, two thirds the size of the word2vec vectors. I also cloned a GitHub repository that implements word2vec in MATLAB, written by Chris McCormick (chrisjmccormick 2017).

It’s important to note the fundamental differences between the two models before discussing the experiment and the results. The word2vec model isn’t as strong as the word2vec embedding *algorithm*. However, I expected the word2vec *embedding* itself to be stronger than the GloVe embedding. This is because word2vec has a vocabulary of around 3 million words and phrases, 2.5 times the size of the GloVe vocabulary. Furthermore, as I already mentioned, the dimension of the GloVe vectors was smaller. These two differences combined lead me to hypothesize that the GloVe model would be less adept at completing analogies. This would not affect the ability to compute Direct Bias, but does say something about the significance of the Direct Bias rating from each dataset.

**Experimental Design**

Word embedding is commonly tested using two of its features: its ability to complete analogies, and general vector operations (which in my case, involved computing Direct Bias).

First, I completed a function for each embedding that completed a list of analogy examples. I wanted to take simple analogies and see how each embedding performed. Then, based on the Bolukbasi paper, I wanted to compute Direct Bias in each dataset on the same set of words, and compare the rating. In *Man is to Computer Programmer as Woman is to Homemaker?* they present an equation for evaluation the Direct Bias:

Where N is the set of words that you want to evaluate bias over (i.e. a set of gender neutral words of one category e.g. types of sports, occupations, college majors etc.). C represents how strict the measure is (e.g. if c is 0, and there is no bias detected between the word and the gender direction g, then the word w has no overlap with g at all).

Finally, g represents “the bias subspace.” For the purposes of this project (as well as in the Bolukbasi paper), g was a gender subspace vector. To form g, you take the vectors corresponding to a set of gender opposite pairs (e.g. man and woman, male and female, waiter and waitress etc.), subtract the male vector from the female vector, and then take the principal components of all the resulting subtractions. A list of the 35 pairs I used to generate g can be found in the **Appendix Figure 7**. Because all the word vectors had been normalized in each dataset, the cosine distance was simply the dot product of the word vector with g, a value between 0 and 1.

I knew that wanted to look at the level of gender based bias in gender neutral occupational titles (e.g. *nurse, doctor, pilot* etc.), because this was used as a measure in the Bolukbasi et al. I compiled a list of 127 gender-neutral occupations by looking through Google’s lists of occupations (“Lists of Occupations” 2017). I ended up with more time on my hands than I had anticipated, so I spoke with Justin Li, who suggested I read some of the articles he consults on how to avoid writing sexist recommendation letters for undergraduates. I ended up consulting a paper published through the Journal of Applied Psychology as a basis for my choice of adjectives (Madera and Michelle R. Hebl and Randi C. Martin 2009). I chose adjectives that had been found frequently within recommendation letters for men and women (around 300 letters). I settled on a list of 96 adjectives. To find the lists of occupations and adjectives, consult the **Appendix**.

**Trends and Takeaways**

*Results*

For the first part of the experiment, I derived a set of analogies that I wanted to test on each model. In the early stages of experimentation, I took very simply analogies and tested the models on them. It became apparent very early on that the GloVe model, trained on Tweets, was much less able to complete analogies than the word2vec model. For this reason, I ended up completing far more analogy examples using the word2vec model, many examples that I didn’t end up testing at all with GloVe because I knew they wouldn’t be completed. Furthermore, word2vec wasn’t just able to complete baseline analogies, but actually returned some very interesting, and sometimes sexist and racist completions to the analogies I had set up.

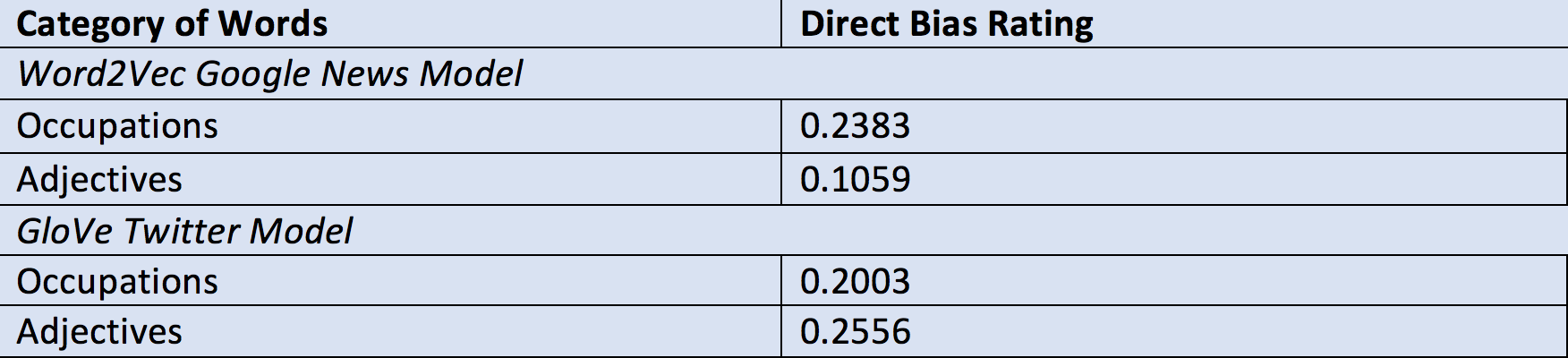
 The second part of the experiment was more focused on vector operations on the vocabulary from each model. First, I generated the gender subspace g, and then computed the Direct Bias for occupations and then for adjectives for each model. The results are found in **Figure 2**.

Figure 2: Direct Bias measures for each model

Figure 2: Direct Bias measures for each model

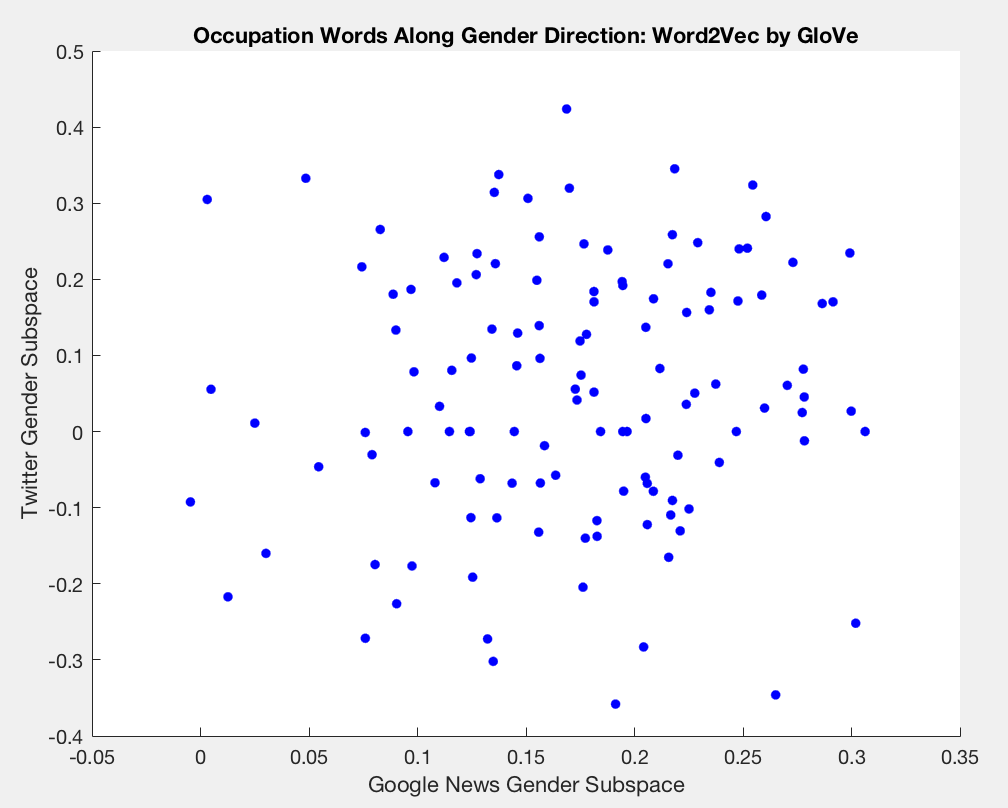
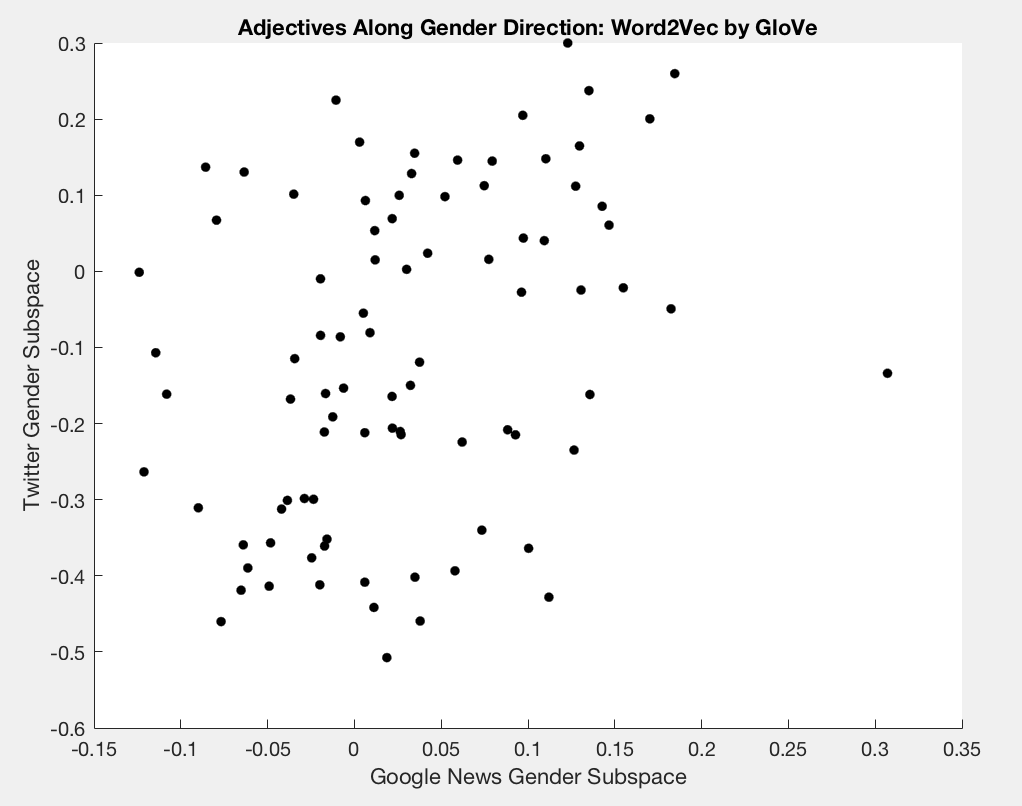
**In the Bolukbasi et al. paper, they also found the distance of each word to the gender subspace for two different word embedding models, to see if the distance was comparable between the two models. This was an ideal measurement in my case because it allowed me to visualize how similar the bias rating was for each word in the occupation or adjective groups between the two models. I also computed Spearman’s rho to see if there existed a correlation between the bias ratings of each dataset. The word distance plot for occupational words is seen in **Figure 3**, and the plot for the adjectives is in **Figure 4**.

Figure 3: Distances between occupational words and gender subspace for each model

I found that for some reason, there was a relatively strong correlation in the case of the adjectives [rho = 0.3609, p < 0.001], but *not* in the case of occupations [rho = 0.1146, p = 0.1995], the p value was too low to reject the null hypothesis.

*Discussion*

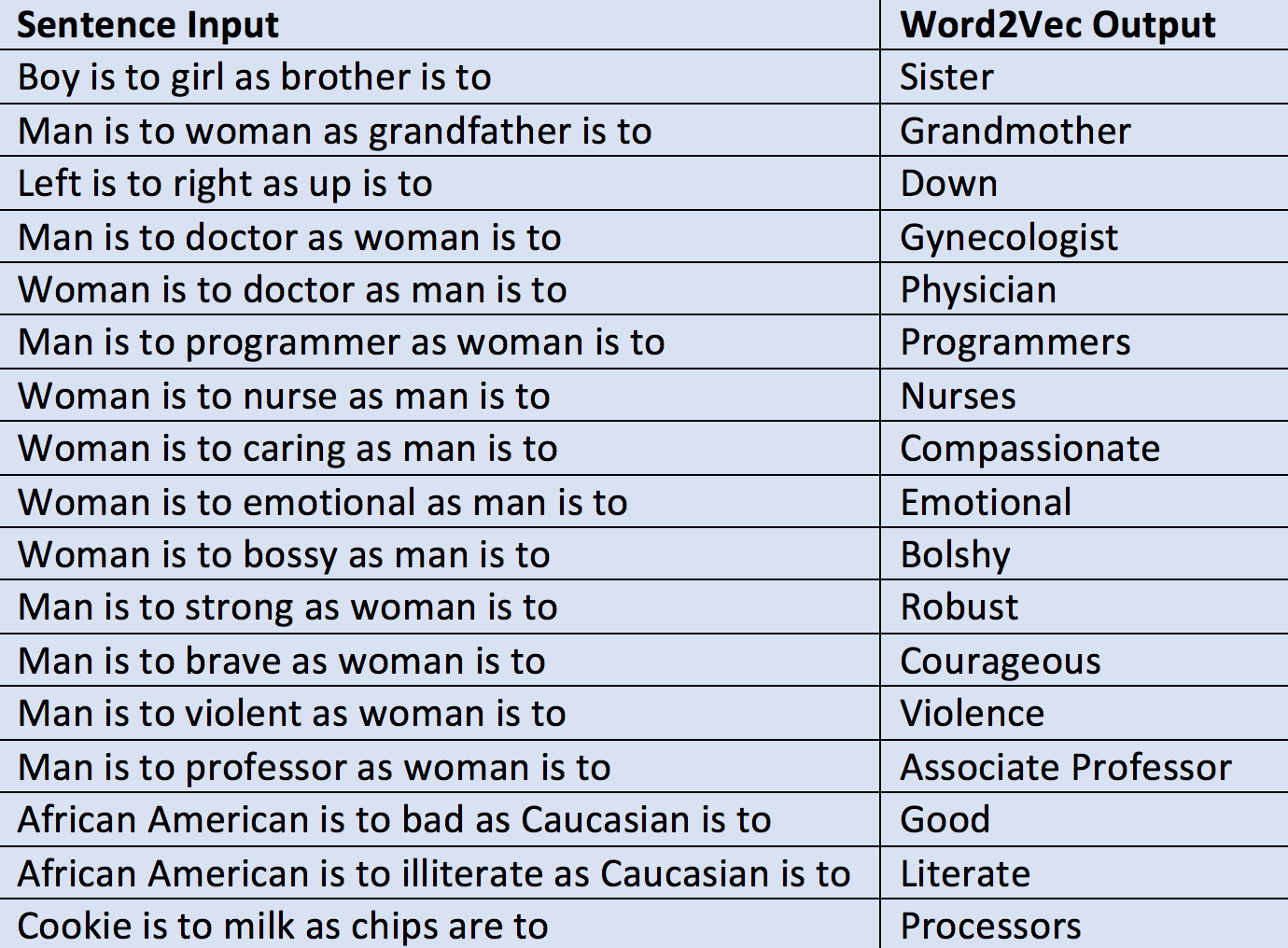
I found the analogy completions particularly illuminating (in the case of word2vec, at least). **Figure 5** shows the analogies from word2vec. In particular, the fourth and fifth analogies are interesting because they display a clear gender bias. I wanted to check the direction, to see if the 5th analogy might return “urologist,” which would at least indicate consistency. The lack of that kind of reciprocity indicates some of the underlying bias the model has learned. The 14th analogy displays the same kind of problem. Finally, I do think it’s important to note the final analogy in **Figure 5**: *Cookie is to milk as chips are to processors*. A student who came to visit me during the computer science showcase came up with this, and I think it displays some of the most fundamental problems with Natural Language Processing in machine learning and artificial intelligence. The word “chip” could signify a snack, a component of a piece of technology, or a means of betting while gambling, and the model has no way of distinguishing those meanings. It’s also funny because the analogy isn’t entirely wrong; cookies “go into” milk much how chips might “go into” a processor.

Figure 4: Distances between adjectives and gender subspace for each model

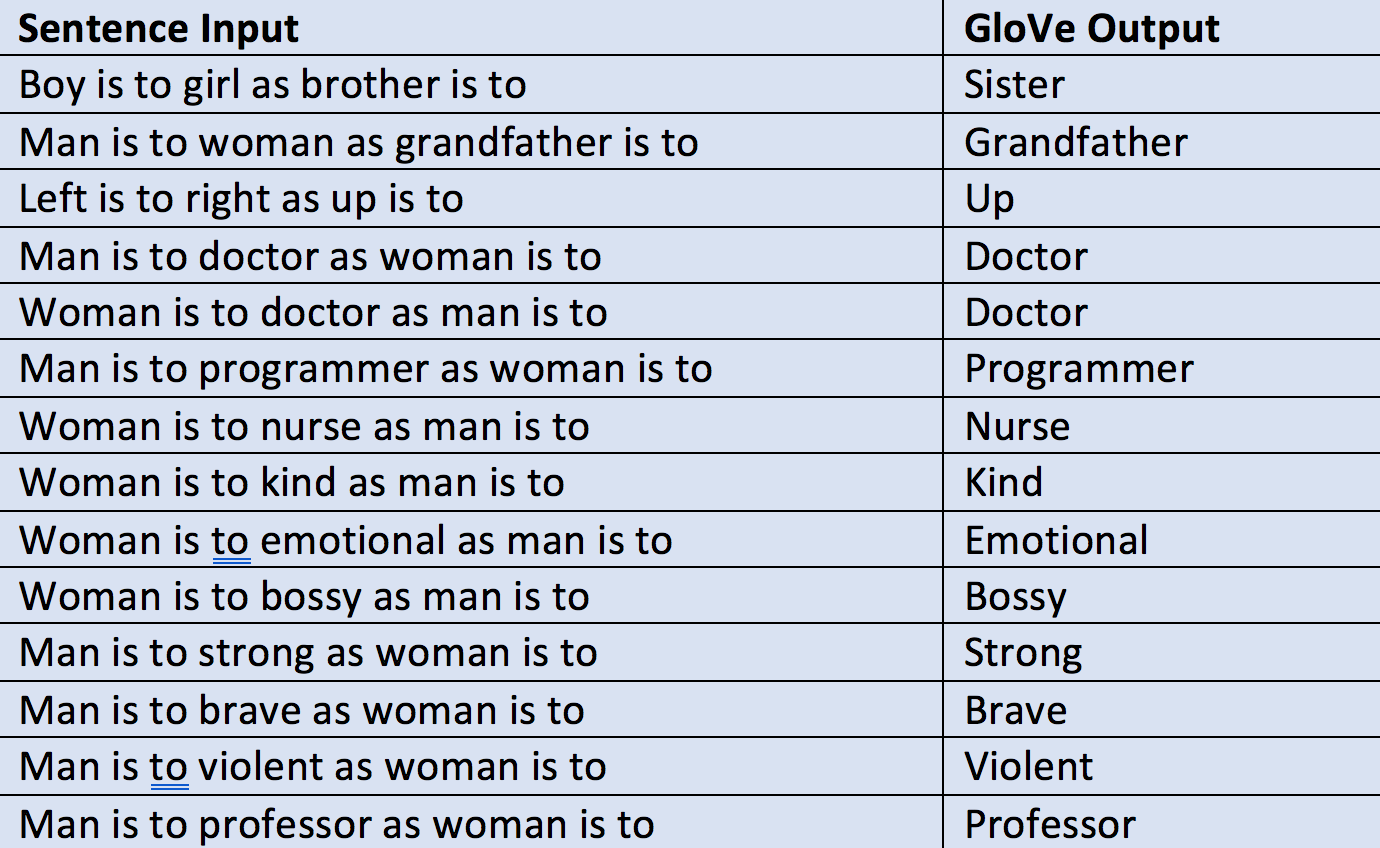
The analogies produced by GloVe were less exciting, as I mentioned earlier. The results of the sample analogies are seen in **Figure 6**. At first glance, it appears as though the model might be miraculously unbiased. It returns the same adjective or noun for each gender. I couldn’t confirm that this was not the case until I had the Direct Bias ratings back, which then confirmed my hypothesis. In the future, choosing a more robust model will be imperative to comparing two models.

Figure 5: Analogy completions from Word2Vec

The most surprisingly takeaway from the Direct Bias ratings is that the rating for adjectives is twice as high in the GloVe model as it is in the word2vec model. This could be due to the differences in size between the two models. If GloVe was trained with a smaller corpus of text, then theoretically, the relationships between pairs of words are stronger because there are fewer unique words. However, this trend does not hold in the case of occupations, so this leads me to believe it is not simply the size of the corpus that makes a difference. Furthermore, it’s interesting that occupation has no correlation between the two models, where adjectives has a very high correlation. This leads me to believe that adjectives are more “set in” to their associations between genders, that is to say, adjectives are more obviously used in the context of one gender than occupations are. Had I obtained a GloVe model trained on a comparable corpus to the word2vec model, I believe the takeaways would be stronger and more generalizable. It could also be that tweet aren’t a good type of text to train a model on. Tweets often involve lots of metaphor, euphemism, and implication. On the other hand, news articles often explain concepts, writing concepts “in plain English.” Intuitively, news seems like a more informative type of corpus than tweets, especially when the end goal is for the model to learn good representation of the semantic meaning of words.

Figure 6: Analogy completions from GloVe

Finally, for fun, I wanted to see the top 10 occupations and adjectives that were closest to the word “male” and the word “female” in each of my datasets. The “top 10s” from word2vec can be found in **Figure 8** and the top 10s from GloVe are in **Figure 9** in the **Appendix**.

**Future Work**

I believe that this project has space for many potential extensions and hopefully, in the spring, I will be able to continue this work as an independent research topic. Both Caliskan and Bolukbasi et al. relied on word embedding as a technique for interpreting the textual data to train the AI on, however, there are other methods for inferring the relationships between words besides word embedding. Future research could model a corpus of text using an alternative NLP method, and then test it for inherited bias. I have also done little exploration in measuring other types of bias, like racism. In the future, I’d like to broaden to other types of bias, and see how strong these various types of bias are, compared to one another.

Personally, this projected offered me a great opportunity to study the intersection between gender and computer science. I am actively involved with Planned Parenthood Club at Occidental, and this project bolsters our mission to create equal access opportunities. Recalling the example from the beginning of the paper regarding gender based advertising, what if an AI does not advertise Planned Parenthood services to a transgender male, just because he has marked “male” as his gender? That individual may benefit or need the services at Planned Parenthood, and miss marketing because of their identity. This problem is prevalent and problematic for all communities. Thus, I believe this research is very important to me, but also to all.

**Appendix**

**Note**: the list of adjectives and occupations used in the Direct Bias analysis are saved in the Documentation folder in “Adjectives.txt” and “Occupations.txt,” respectively.

Figure 7: Gender pairs used to derive gender subspace

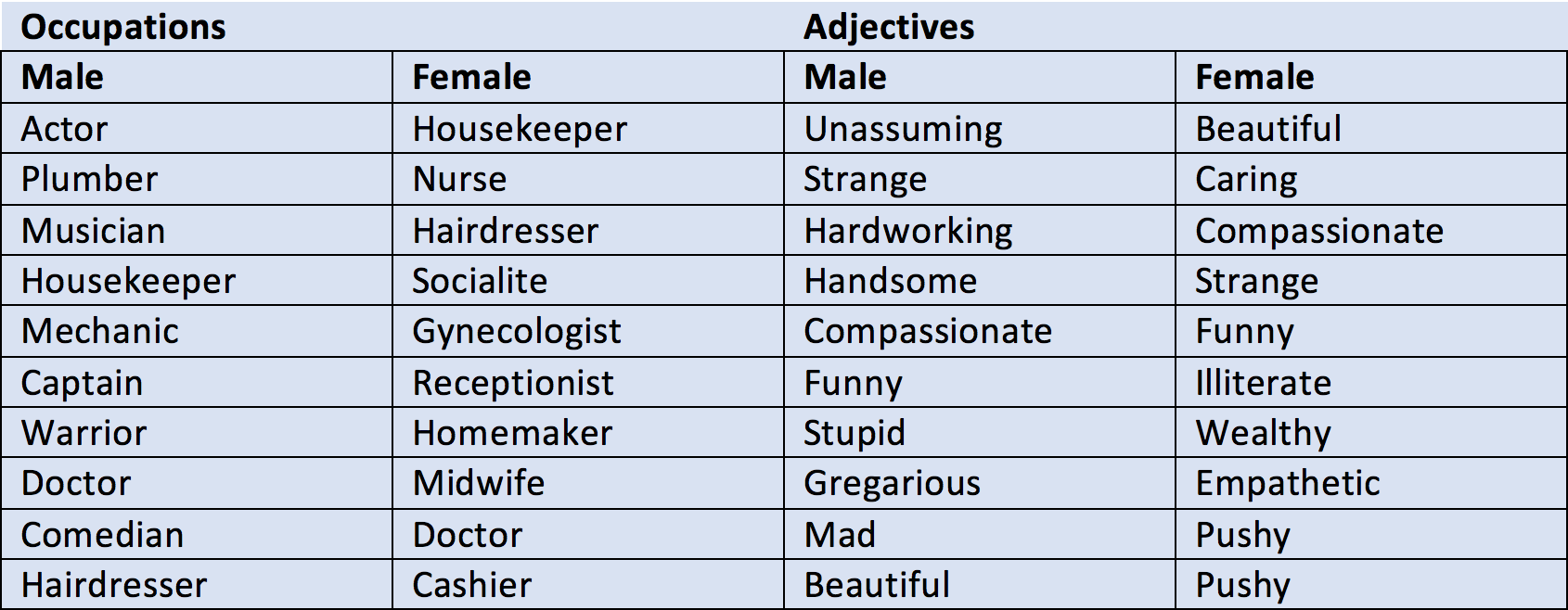


Figure : Top 10s from Word2Vec

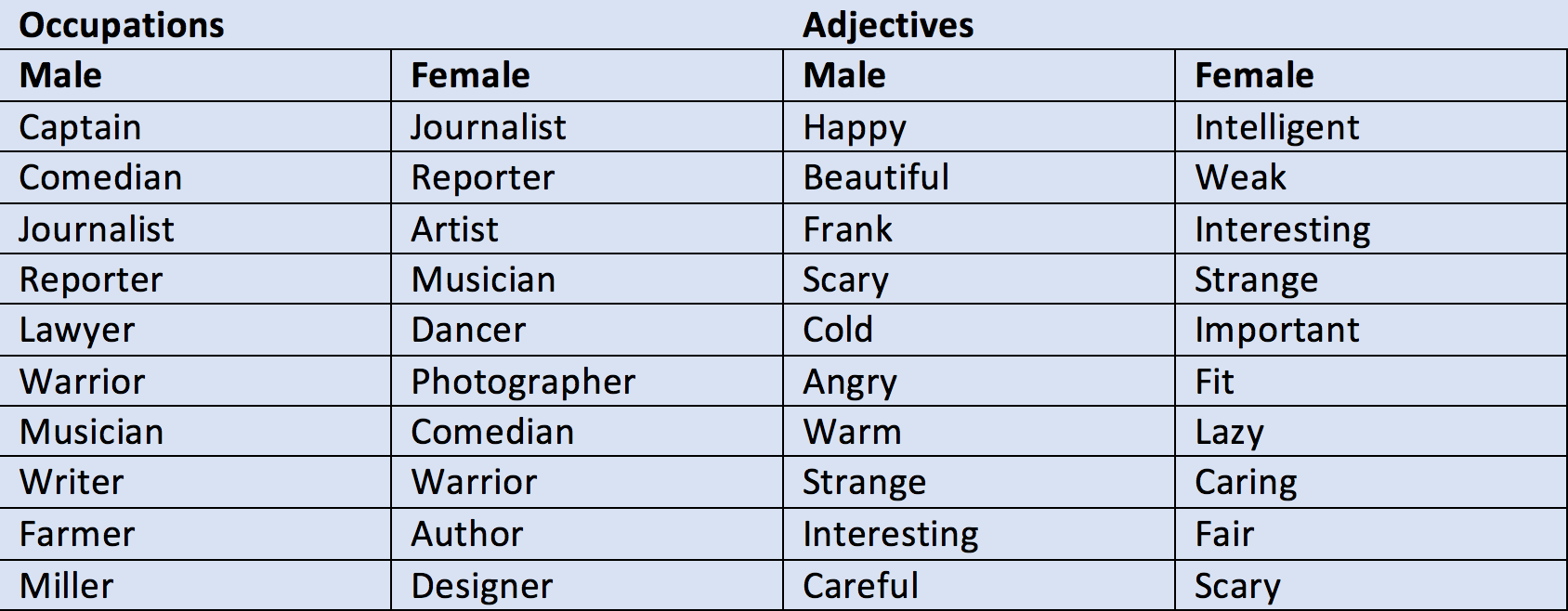


Figure : Top 10s from GloVe

**References:**

chrisjmccormick. (2016) 2017. *Contribute to Word2vec\_matlab Development by Creating an Account on GitHub*. Matlab. https://github.com/chrisjmccormick/word2vec\_matlab.

Datta, Amit, Michael Carl Tschantz, and Anupam Datta. 2015. “Automated Experiments on Ad Privacy Settings.” *Proceedings on Privacy Enhancing Technologies* 2015 (1):92–112. https://doi.org/10.1515/popets-2015-0007.

“Lists of Occupations.” 2017. *Wikipedia*. https://en.wikipedia.org/w/index.php?title=Lists\_of\_occupations&oldid=800344985.

Madera, Juan M., and Michelle R. Hebl and Randi C. Martin. 2009. “Gender and Letters of Recommendation for Academia: Agentic and Communal Differences.” *Journal of Applied Psychology* 94 (6):1591–99.

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. 2014. “GloVe: Global Vectors for Word Representation.” In *Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. http://www.aclwe

b.org/anthology/D14-1162.

Simonite, Tom. 2017. “Machines Taught by Photos Learn a Sexist View of Women.” News. WIRED. August 21, 2017. https://www.wired.com/story/machines-taught-by-photos-learn-a-sexist-view-of-women/.

Tomas Mikolov. 2013. “Google Code Archive - Long-Term Storage for Google Code Project Hosting.” Google Code Archive. July 29, 2013. https://code.google.com/archive/p/word2vec/.