Surveying Bias in Word Embeddings of Political News Media Dataset

INFO I 501 Introduction to Informatics

Luddy School of Informatics , Computing and Engineering at Indiana University

Ming Jiang

May 3 , 2023

Group 4

Soumya Shanigarapu , Supraja Pericherla , Durga Pravallika Kuchipudi

LUDDY
SCHOOL OF INFORMATICS,
COMPUTING, AND ENGINEERING

1. ABSTRACT

In the age of machine learning, word embeddings have become a powerful tool for natural language processing (NLP), allowing the encoding of word semantics in real-valued vectors. However, while word embeddings have revolutionized NLP, they are not without flaws. They are often trained on large datasets that reflect biases and prejudices present in our society, leading to models that perpetuate and amplify social biases. This has raised concerns about bias and discrimination in NLP systems, prompting researchers to develop techniques for detecting and mitigating biases in word embeddings.

One area of study in AI fairness is the impact of political leanings on word embedding bias. In this project, we aim to investigate the presence of biases in word embedding models trained on "liberal" and "conservative" news articles and evaluate the models' performance on various bias metrics such as gender, religion, race, gender identity and age . By comparing the biases present in the two models, we hope to gain insights into the factors that contribute to bias in word embeddings and develop more equitable NLP techniques for future applications.

Our investigation is particularly timely given the polarized nature of contemporary politics and media, where opposing views are often presented through different lenses and frames. The language we use can shape our perceptions and biases, and word embeddings have the potential to amplify these biases. By promoting fairness, transparency, and accountability in NLP systems, we can build a more equitable future for all.

As language barriers pose significant challenges for communication and collaboration in an increasingly interconnected world, this research can contribute to the development of more inclusive and fair NLP systems that can help bridge these barriers. By understanding the complex relationship between language, politics, and biases in machine learning, we can develop more responsible and equitable use of natural language processing technologies

TABLE OF CONTENTS

2.1 INTRODUCTION

Word embeddings have become a ubiquitous tool in natural language processing (NLP) that encodes word semantics as real-valued vectors. The idea behind word embeddings is that the meaning of a word can be represented by the words that appear in its vicinity, such that words with similar meanings are assigned similar vectors. For instance, the word vectors for "king" and "queen" should be closer together than the vectors for "king" and "banana". Word embeddings have been used to enhance various NLP tasks such as text classification, sentiment analysis, and machine translation.

However, word embeddings are not without their limitations. One of the biggest challenges with word embeddings is that they are known to contain biases that can perpetuate harmful social stereotypes. For instance, the word "doctor" might be more closely associated with men than women in a biased word embedding, due to historical gender imbalances in the medical profession.

Recent research has shown that word embeddings can also be influenced by the political leanings of news sources. A particular study found that word embeddings trained on news articles from liberal sources were more likely to associate words like "corporation" with negative connotations, while embeddings trained on conservative sources were more likely to associate words like "immigrant" with negative connotations. These findings suggest that political ideology can impact the way language is represented in machine learning models, which could have significant implications for applications that rely on these models.

In this project, we aim to investigate the relationship between political ideology and biases in word embeddings. Specifically, we will compare word embeddings trained on corpora of "liberal" and "conservative" news articles using popular algorithms such as Fast Text and Word2Vec. We will analyze various types of social biases in the embeddings, such as gender bias, religion bias,racial bias,gender identity bias and age bias to determine whether the political leanings of news sources contribute to biases in the representations of language. Ultimately, our goal is to raise awareness about the potential risks of using biased word embeddings and promote the development of fairer and more equitable NLP techniques.

2.2 WORD EMBEDDING BIAS

As the world becomes increasingly interconnected and globalized, language barriers pose a significant challenge for communication and collaboration. Natural Language Processing (NLP) technologies can help overcome these barriers by enabling machines to understand and process human language. One of the key components of NLP is word embeddings, which represent words as numerical vectors in a high-dimensional space. These vectors can capture the meanings and relationships between words, making them useful for a wide range of tasks such as language modeling, sentiment analysis, and machine translation.

However, the use of word embeddings has also raised concerns about the potential for bias and discrimination in NLP systems. The biases in word embeddings can stem from a variety of sources, such as historical stereotypes, cultural norms, and demographic imbalances in training data. For example, word embeddings trained on a corpus of news articles may associate certain races or ethnicities with negative words or concepts, perpetuating harmful stereotypes.

To address these concerns, researchers have developed a variety of techniques to detect and mitigate bias in word embeddings, such as debiasing algorithms and fairness metrics. These methods aim to make NLP systems more equitable and inclusive, by promoting fairness, transparency, and accountability. However, the effectiveness of these techniques may vary depending on the nature and complexity of the biases in the training data, and more research is needed to develop robust and scalable methods for bias mitigation in NLP systems.

In this project, we aim to examine how political affiliations can influence bias in word embeddings. Specifically, we plan to compare the biases present in two word embedding models trained on "liberal" and "conservative" news articles, respectively, with a focus on evaluating the models' performance on various bias metrics, such as gender, race, and religion. By using methods that distinguish between liberal and conservative texts, we hope to gain insights into the factors that contribute to bias in word embeddings and develop more equitable NLP techniques for future applications.

3.LITERATURE REVIEW

[1] Bolukbasi, T., Chang, K.-W., Zou, J., Saligrama, V., and Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? debiasing word embeddings. arXiv preprint arXiv:1607.06520.discusses the problem of gender bias in word embeddings, which are vector representations of words used in natural language processing. The authors show that commonly used word embedding algorithms can reflect and even amplify gender biases in language, leading to biased results in NLP applications. They propose a debiasing method to remove gender associations from word embeddings and show that it improves performance on gender-neutral tasks.

[2] "All the News 2.7 Million News Articles and Essays" is a dataset of news articles and essays collected from over 7,000 sources, spanning a period of three years from 2015 to 2018. The dataset includes text, publication date, author, and other metadata. It has been used for various research purposes, including training language models and analyzing media bias.

[3] "All the News: 143,000 Articles from 15 American Publications" is a similar dataset of news articles collected from 15 American publications over a period of three years from 2014 to 2017. The dataset includes text, publication date, author, and other metadata. It has also been used for various research purposes, including studying media bias and analyzing changes in language use over time.

[4] "Quantifying 60 Years of Gender Bias in Biomedical Research with Word Embeddings" by Rios et al. (2020) analyzes gender bias in the language used in biomedical research papers over a period of 60 years. The authors use word embeddings to quantify gender associations with different professions and find that biomedical research has been historically biased towards male-associated professions. They also show that recent trends towards gender-neutral language use in biomedical research are reducing this bias.

[5] Malvina Nissim, Rik van Noord, and Rob van der Goot. "Fair is better than sensational: Man is to doctor as woman is to doctor". arXiv preprint arXiv:1905.09866, 2019. analyzes gender bias in word embeddings and proposes a method for debiasing them. The authors show that commonly used word embedding algorithms reflect gender biases and that these biases can affect downstream applications. They propose a method that modifies the embeddings to reduce gender bias while preserving other semantic relationships between words.

[6] A. Caliskan, J. J. Bryson, and A. Narayanan." Semantics derived automatically from language corpora contain humanlike biases". Science, 356(6334):183–186, 2017 analyzes biases in language using a large dataset of text. The authors show that certain associations between words, such as between gender and career or between ethnicity and crime, are present in language corpora and reflect human biases. They argue that this can have implications for machine learning applications that rely on language models and propose methods for reducing bias.

4.METHODOLOGY:

4.1 METHODOLOGY FLOWCHART

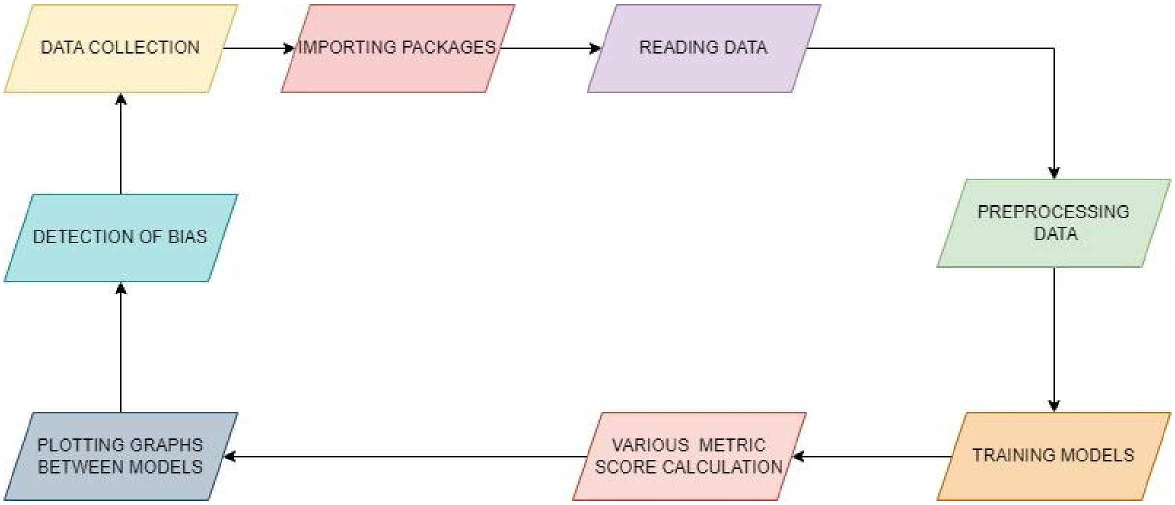


Fig 1.1 Flowchart Methodology

4.2 STEPS IN CONSTRUCTING WORD EMBEDDING MODEL

4.2.1 DATA COLLECTION

After finalizing a topic for our project study and conducted a thorough search for relevant data from various sources such as Kaggle , Github and academic journals. We were able to obtain two distinct datasets from the Github website which will be used for our analysis. The datasets are as follows. Link

4.2.2 DATA DESCRIPTION

We were fortunate to acquire a preprocessed dataset for our project, which consists of news articles from both "liberal" and "conservative" media sources. This dataset was obtained from a well-known publication and research group called Components.

To compile the dataset, Components scraped articles from various media sources, including CNN, Washington Post, Buzzfeed News, Fox News, Breitbart, and New York Post. The liberal corpus contained 30,000 articles from CNN, Washington Post, and Buzzfeed News, while the conservative corpus contained 12,229 articles from Fox News and 30,000 articles from both Breitbart and New York Post.

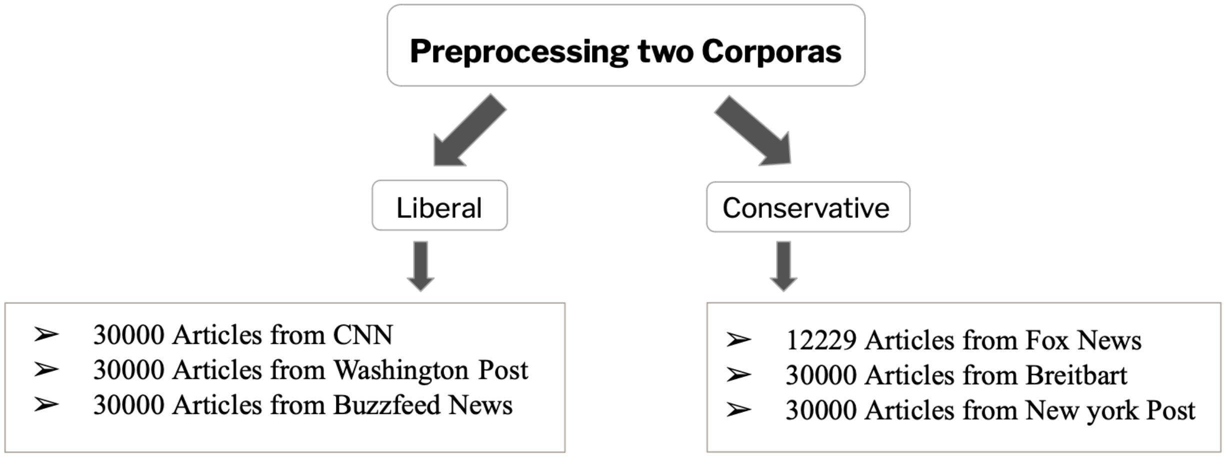

Preprocessing two Corporas
Liberal
Conservative
➢ 30000 Articles from CNN
➢ 30000 Articles from Washington Post
➢ 30000 Articles from Buzzfeed News
➢ 12229 Articles from Fox News
➢ 30000 Articles from Breitbart
➢ 30000 Articles from New york Post

Fig 1.2 Dataset flowchart

It's worth noting that the articles in the dataset were collected from 2016 to 2020, ensuring that we have recent news articles. Additionally, the dataset underwent some preprocessing steps such as tokenization, stop word removal, and lemmatization, which makes it more manageable for us to use for our project without having to spend extra time cleaning and processing the raw data ourselves.

- **Liberal dataset:** Text data from liberal or left-leaning media sources, used to study progressive views.
- **Conservative dataset**: Text data from conservative or right-leaning media sources, used to study conservative views.

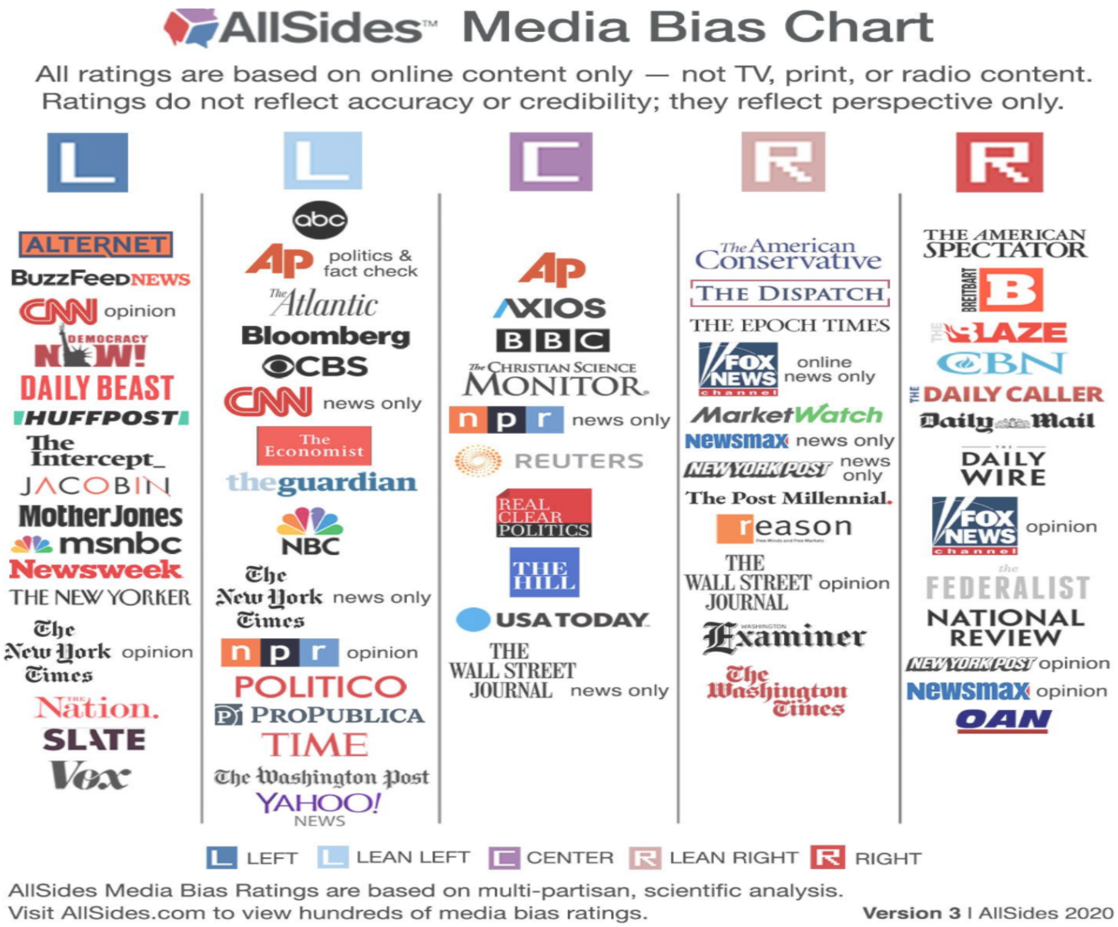


Fig 1.3 Figure showing the various types of media that fall under the categories of "conservative" and "liberal" datasets

4.2.3 WORD CLOUD

In order to identify highly repeated words in a large dataset containing nearly 13 million words, a word cloud analysis was employed. This approach helps us visualize the most frequently occurring words in datasets. Word clouds can be used to identify potential biases in text data by visualizing the frequency and distribution of words within the text.

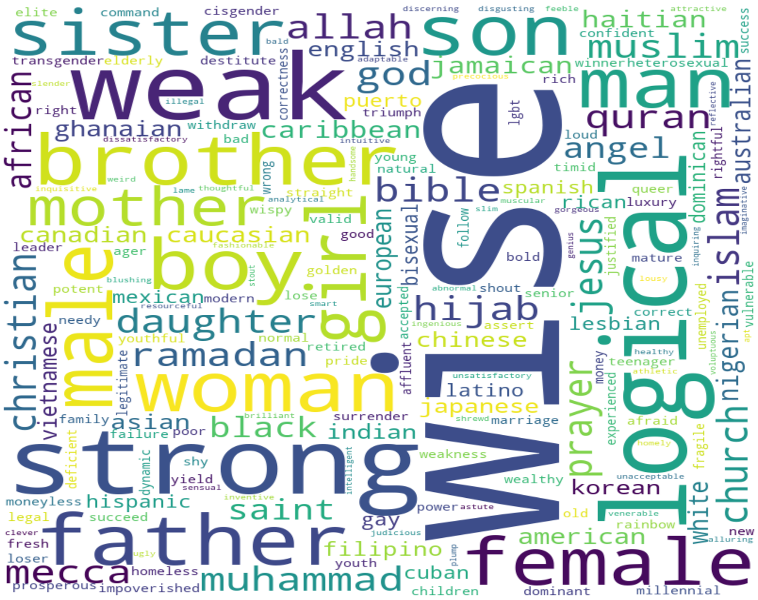


Fig 1.4 Visual representation of the most frequently repeated words within the dataset

4.3 CHOOSING WORD EMBEDDING MODELS

In our study, we aimed to compare biases among different language models such as fasttext, bert, glove, and word2vec. However, due to issues with improper datasets and training difficulties, we were only able to successfully train fasttext and word2vec models. To visualize the frequent occurrence of words in our datasets, we used word cloud. Training the models took a significant amount of time, and we were unable to generate visualizations for bert and glove models due to long datasets loading time. Despite these limitations, we were able to compare biases and find metrics for fasttext and word2vec models.

4.4 METHODOLOGIES FOR WORD2VEC & FASTTEXT MODELS

4.4.1 IMPORTING LIBRARIES FOR WORD2VEC & FASTTEXT

1. **FastText**: FastText is a library for efficient learning of word representations and sentence classification developed by Facebook AI Research. You can install it using pip by running the command "pip install fasttext".
2. **NumPy:** NumPy is a Python library for numerical computing that provides support for arrays and matrices. You can install it using pip by running the command "pip install numpy".
3. **Matplotlib**: Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. You can install it using pip by running the command "pip install matplotlib". Alternatively, you can also use the command "!pip install matplotlib" if you are running the code in a Jupyter notebook.
4. **mpl_toolkits**: The mpl_toolkits module contains various tools to help with matplotlib's functionality. You can install it using pip by running the command "pip install mpl_toolkits".
5. **Gensim:** Gensim is a Python library for topic modeling, document indexing, and similarity retrieval with large corpora. You can install it using pip by running the command "pip install gensim".
6. **CSV**: CSV is a built-in module in Python for reading and writing CSV (Comma-Separated Values) files. No pip install is required as it is part of Python's standard library.
7. **WEAT**: WEAT is a feature of the WEFE (Word Embedding Fairness Evaluation) library. To install WEFE and its dependencies, use the following pip command: "pip install wefe[all]".
8. **Math**: Math is a built-in module in Python for mathematical operations. No pip install is required as it is part of Python's standard library.

Encoding: When reading a file, you may need to set the encoding to match the file's format. In this case, the encoding used is 'ISO-8859-1'.

4.4.2WORD2VEC METHOD:

We employed the Word2vec approach while leveraging the Python module gensim to construct our word embedding models. All the words in our corpus were lowercased for preprocessing, and we used the NLTK package to tokenize them into sentences of word tokens needed by genism.

By assessing the correlation between two sets of target concepts and two sets of qualities, we employed the Word Embedding Association Test (WEAT) developed by Caliskan [6] to identify implicit bias. To investigate strong and weak bias in different groups, we have aimed to assess the degree of bias in the representation and recognition of individuals across various domains, including gender, race, and religion. This will involve collecting data on the breakdown of individuals in these groups in different domains, as well as examining any differences in representation and opportunities received by individuals in each group.

The attributes would consist of a group of terms that describe "male," "man," "boy," "brother," "king," etc., and a group of words that describe "female," "woman," "girl," "sister," "queen," etc.We are looking at strong and weak notions as the target concepts, therefore a group of terms for strong might be "power," "strong," "confident," "dominant," "potent," etc., while a set of words for weak may be "weak," "surrender," "timid," "vulnerable," "weakness," etc. In short, the WEAT attempts to summarize whether terms connected to men are more likely to be strong terms than terms related to women (i.e., whether phrases related to women are more likely to be weak terms than terms related to men).

CALCULATION OF WEAT SCORE:

Formally, the calculations are done with the equation in the figure below.

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

$$s(w,A,B) = \left[\sum_{a\in A} \cos(\vec{\mathbf{w}},\vec{\mathbf{a}})\right] - \left[\sum_{b\in B} \cos(\vec{\mathbf{w}},\vec{\mathbf{b}})\right],$$

Fig 1.5 WEAT Score Formula

A,B refers to the traits like male and female, whereas X,Y refers to the target sets, such as strong and weak. We compute s(w, A, B) and compare the difference of total for each word in the target sets X and Y, such as "power" vs. "weak." By comparing the target word's cosine similarity to each word in the attribute set, such as "boy" vs. "girl," and taking the difference of the sum, s(w, A, B) is determined. The values normally fall between -2 and 2, with 0 denoting no bias. A positive number in the formula for X, Y, A, and B denotes that X is nearer A (and vice versa, that Y is nearer B). A negative value means that Y is closer to A than X is to B.

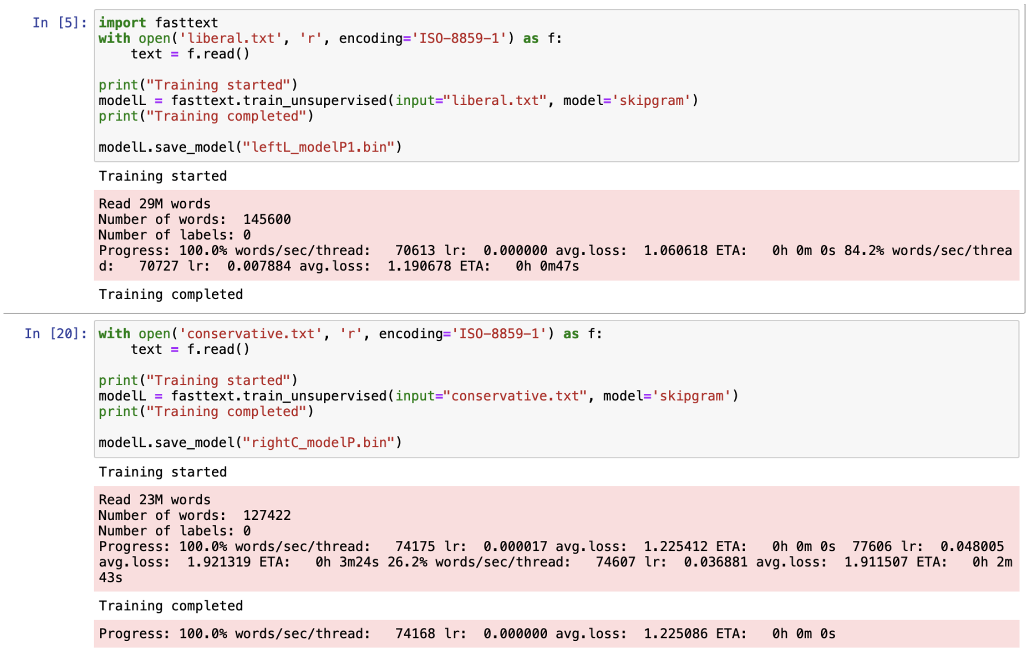
WEAT was computed using the Python WEFE (The Word Embedding Fairness Evaluation Framework) package . For our analysis, we looked at the pairs of the following attribute sets: (male, female), (Islam, Christian), (White, Black), (White, Asian), (LGBTQ, Straight), and (Old, Young). We examined the target set pairs for the neutral terms (strong vs weak), (normal, abnormal), and (intelligence, appearance).

4.4.3FASTTEXT METHOD:

4.4.3.1`WEAT Score in Fasttext Model:

i) Training the Models in Fasttext:

We used FastText's unsupervised training with skipgram architecture to generate word embeddings that capture relationships between words in our text corpus without the need for labeled data. Additionally, ISO-8859-1 encoding was used over UTF-8 encoding, as the former has a smaller file size and can efficiently represent characters in Latin-based languages, while the latter is necessary to represent characters in non-Latin scripts and avoid data loss or corruption.



```
In [5]: import fasttext
        with open('liberal.txt', 'r', encoding='ISO-8859-1') as f:
            text = f.read()

        print("Training started")
        modelL = fasttext.train_unsupervised(input="liberal.txt", model='skipgram')
        print("Training completed")

        modelL.save_model("leftL_modelP1.bin")

        Training started

        Read 29M words
        Number of words:  145600
        Number of labels: 0
        Progress: 100.0% words/sec/thread:   70613 lr:  0.000000 avg.loss:  1.060618 ETA:   0h 0m 0s 84.2% words/sec/threa
        d:   70727 lr:  0.007884 avg.loss:  1.190678 ETA:   0h 0m47s

        Training completed

In [20]: with open('conservative.txt', 'r', encoding='ISO-8859-1') as f:
             text = f.read()

         print("Training started")
         modelL = fasttext.train_unsupervised(input="conservative.txt", model='skipgram')
         print("Training completed")

         modelL.save_model("rightC_modelP.bin")

         Training started

         Read 23M words
         Number of words:  127422
         Number of labels: 0
         Progress: 100.0% words/sec/thread:   74175 lr:  0.000017 avg.loss:  1.225412 ETA:   0h 0m 0s  77606 lr:  0.048005
         avg.loss:  1.921319 ETA:   0h 3m24s 26.2% words/sec/thread:   74607 lr:  0.036881 avg.loss:  1.911507 ETA:   0h 2m
         43s

         Training completed

         Progress: 100.0% words/sec/thread:   74168 lr:  0.000000 avg.loss:  1.225086 ETA:   0h 0m 0s
```

Fig 1.6 Training models in Fasstext

In summary, using the unsupervised training method and skipgram architecture in FastText helps to generate word embeddings that capture the semantic and syntactic relationships between words in our text corpus without the need for any labeled data or supervision.

ii) Importing Pre trained Fasttext Models & Bias Metrics to Calculate:

We loaded our two pre-trained FastText models, one for liberal words and the other for conservative words. It uses the `load_facebook_vectors` function from Gensim to load the word vectors from these models. Then, it creates two instances of the WordEmbeddingModel class from the wefe library, one for each model. These instances are used to calculate bias metrics using the WEAT method, which compares the similarities between word embeddings of different word sets and measures the degree of association between each set and a target attribute.

iii) Performing WEAT Score :

This function calculates the WEAT (Word Embedding Association Test) score for a given set of groups, target terms, languages, and a word embedding model. The WEAT score measures the degree of association between the target terms and the groups in a given model. A positive WEAT score indicates that the model shows a bias towards the target terms and groups, while a negative score indicates the opposite.

We took the same word sets and groups and metrics that we used for the word2vec model; it is easy to identify the bias metrics between these two models.

Cohen’s D in Fasttext Model:

Cohen's d is a statistical measure used to quantify the difference between two groups in terms of standard deviation units. It is calculated by taking the difference between the means of two groups and dividing it by the pooled standard deviation of the two groups. A Cohen's d of 0.2 is considered a small effect size, 0.5 a moderate effect size, and 0.8 or higher a large effect size. Cohen's d is commonly used in statistical analysis to assess the magnitude of a difference between two groups, and to evaluate the effectiveness of interventions or treatments.

$$Cohen's\ d_s = \frac{M_2 - M_1}{Pooled\ SD}$$

$$Pooled\ SD\ = \sqrt{\frac{(n_1 - 1) \times SD_1{}^2 + (n_2 - 1) \times SD_2{}^2}{n_1 + n_2 - 2}}$$

Fig 1.7 Cohen’s D Formula

The formula is applied separately for each combination of gender and association type (strong or weak). The means and standard deviations of the word vectors for each group are first calculated. Then, the pooled standard deviation is calculated using the means and standard deviations of both groups. Finally, Cohen's d is computed as the difference between the means of the two groups divided by the pooled standard deviation.

We performed Cohen’s D for the same word sets and groups we have previously used for word2vec and fasttext WEAT Score.

5.Results:

Based on our WEAT metric scores we have plotted Strong vs Weak,Intelligence vs Appearance,Normal vs Abnormal which are target word pairs against the attributes such as gender group,religion group,race group,gender identity group and age groups.Our finding are graphed as below.

1. **Strong vs Weak w.r.t gender,religion,race,gender identity and age groups:**

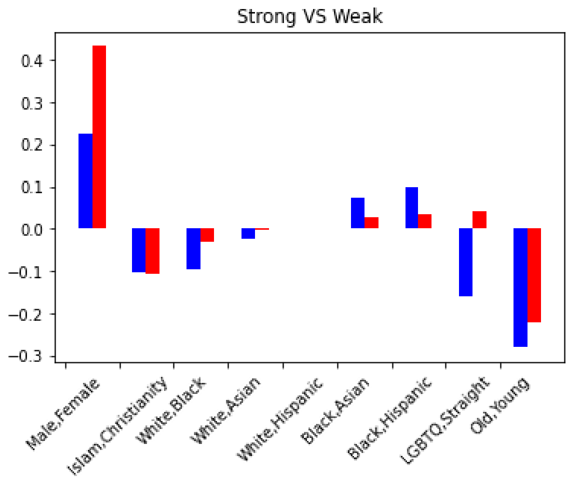


Fig 5.1:Strong and Weak vs biased groups

2.Intelligence vs Appearance w.r.t gender,religion,race,gender identity and age groups:

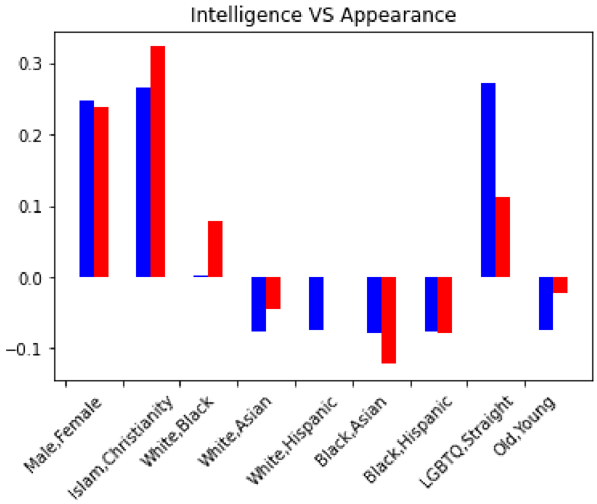


Fig 5.2:Intelligence and Appearance vs biased groups

3.Normal vs Abnormal w.r.t gender,religion,race,gender identity and age groups:

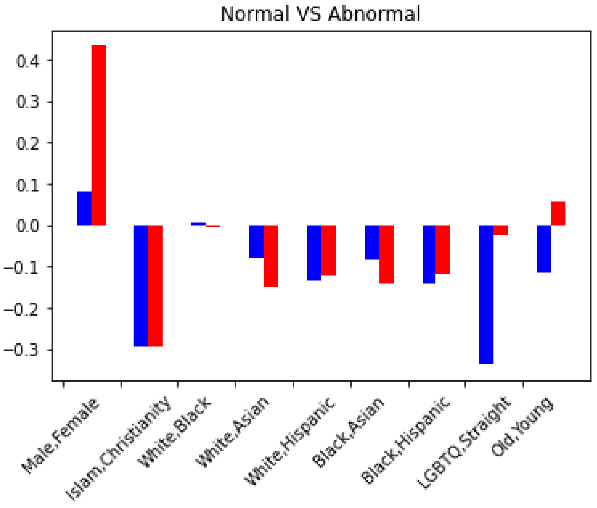


Fig 5.3:Normal and Abnormal vs biased groups

The blue line refers to the liberal model, and the red line refers to the conservative model. For interpretation, consider figures 1,2,3,

Comparing "male and female" with respect to "intelligence and appearance", both the liberal and conservative model gave a similar positive WEAT score, indicating that male terms are closer to intelligence terms (i.e. female terms are closer to appearance terms).
In general, the degree of bias or the absence of bias were similar between liberal and conservative models. Still, there were a couple of analogies where we could see some difference.

For instance, Consider the normal vs abnormal graph. Normal terms include words like "natural", "right", "normal", while abnormal terms include terms like "weird", "abnormal", "wrong". For male and female pairs, the conservative model had a much higher score, suggesting that male is closer to normal, and female is closer to abnormal. On the other hand, for LGBTQ and straight, the liberal had a much more negative score, suggesting that LGBTQ is closer to abnormal, and straight is closer to normal. The limitation of the WEAT score is that it cannot be interpreted directly. Like the co- sine similarity, it is only a comparison measure.

The conservative model is represented by the red line, and the liberal model is represented by the blue line. Both the liberal and conservative models produced identical positive WEAT scores when "male and female" were compared to "intelligence and appearance," showing that male terms are more closely related to intelligence terms (and female terms are more closely related to terms of appearance).

Between liberal and conservative models, there was generally little difference in the level of prejudice or its absence. However, there were a few analogies where we could distinguish some differences. Consider the normal vs. abnormal graph as an illustration. Normal terms include adjectives such as "natural," "right," and "normal," while abnormal terms include adjectives such as "weird," "abnormal," and "wrong". The conservative model scored significantly higher for the male and female pair, indicating that the man is closer to normal while the female is closer to abnormal. In contrast, the liberal had a substantially lower score for both LGBTQ and straight people, indicating that the former group is more anomalous than the latter. The WEAT score's drawback is that it cannot be immediately comprehended. It serves merely as a comparison metric, like the co- sine similarity.

2) Fasttext - WEAT Score :

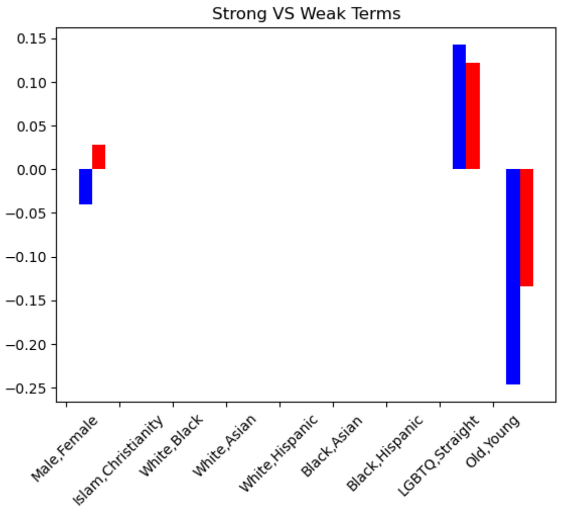


Fig 1.7 Strong vs Weak vs Gender , Religion , old vs young categories etc..,

For the Male and Female and Strong and Weak pair, both liberal and conservative models had negative WEAT scores, but the conservative score was closer to zero, suggesting a weaker association. This may suggest a subtle gender bias across political ideologies.For the LGBTQ and Straight and Strong and Weak pair, both models had positive WEAT scores, but the liberal score was higher, indicating a stronger association between LGBTQ words and weakness in the liberal model. This suggests that the association between the LGBTQ community and weakness may be stronger among politically liberal individuals.For the Old and Young and Strong and Weak pair, both models had negative WEAT scores, but the liberal score was much more negative, indicating a stronger association between youth and strength among politically liberal individuals. This suggests a generational divide in attitudes towards aging and strength.

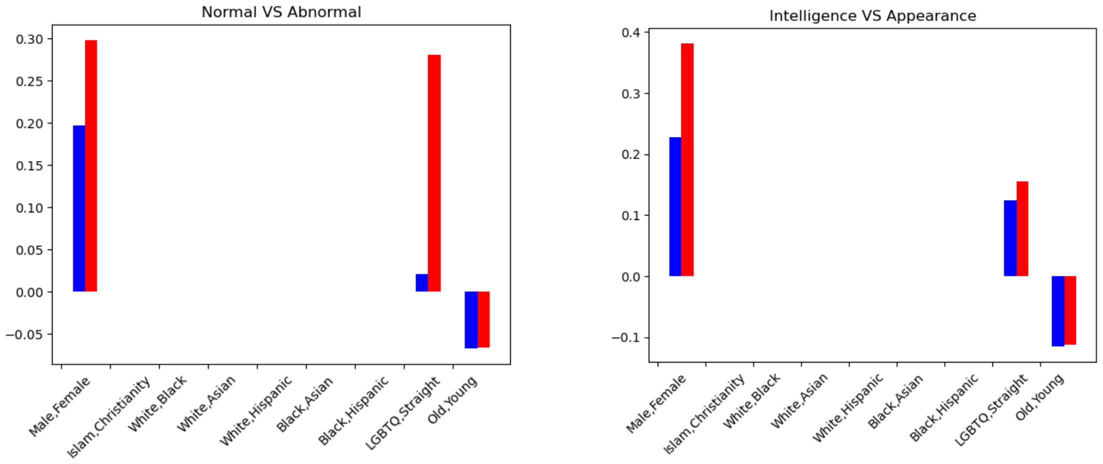


Fig 1.8 Normal vs Abnormal ,and Intelligence vs Appearance w.r.t Religion , old vs young categories etc..,

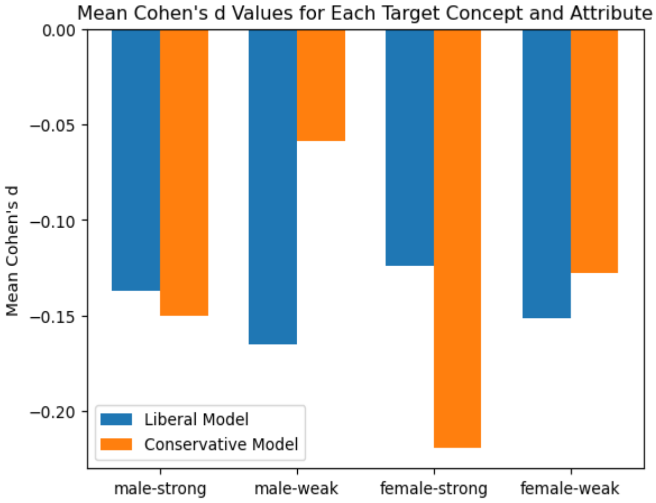
Male and Female: Liberal model shows similar association between males and females with normal and abnormal, whereas conservative model suggests a different association.LGBTQ and Straight: Liberal model shows similar association between LGBTQ and straight with normal and abnormal, whereas conservative model suggests a stronger association between LGBTQ and abnormality.Old and Young: Liberal model shows similar association between old and young with normal and abnormal, whereas conservative model suggests a slightly stronger distinction between these age groups.

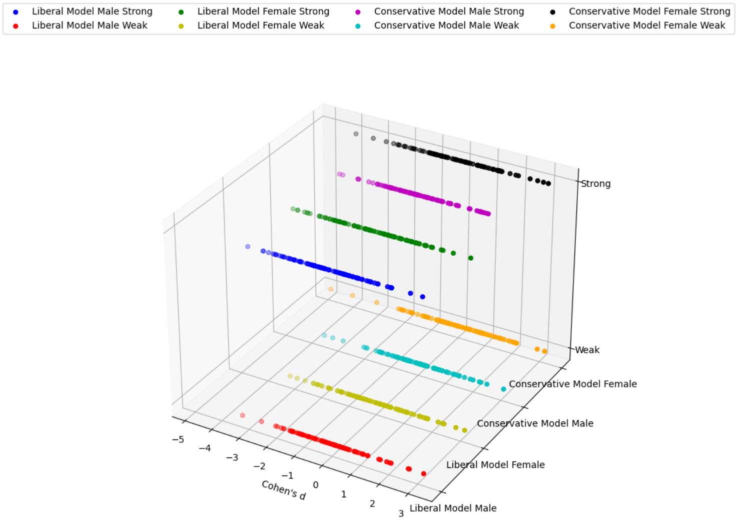
For Male and Female wrt Intelligence and Appearance, both L and C models show a stronger association between males and intelligence and females and appearance, with the association being stronger in the C model. For LGBTQ and Straight and Old and Young wrt Intelligence and Appearance, both L and C models show a stronger association between straight/young individuals and intelligence and LGBTQ/old individuals and appearance, but the strength of this association is weaker overall.

Here We can observe that the scores were NaN (not a number) for both of our trained fastText Liberal and Conservative models. This can occur if the pair of word sets in the WEAT is not present in the trained vocabulary of the models. It is important to note that this can happen and is not a reflection of any issues with our models or methodology.

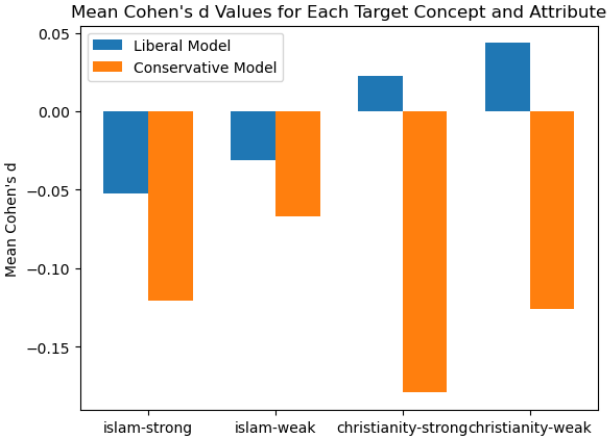
3) **Fasttext - Cohen’s D Value :**

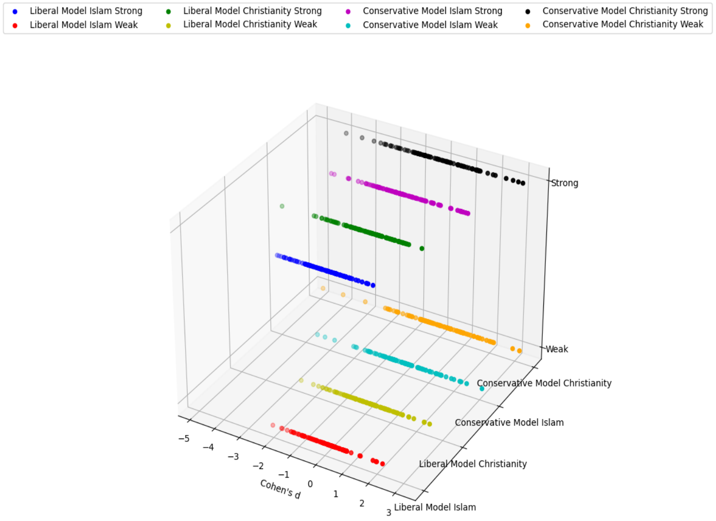
i) Strong vs Weak w.r.t Male vs Female



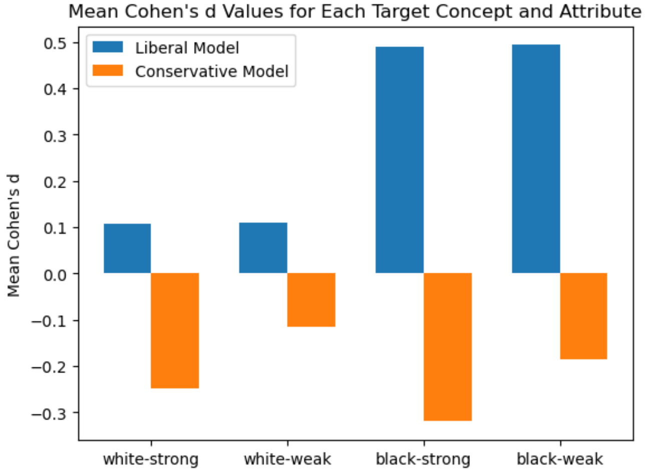


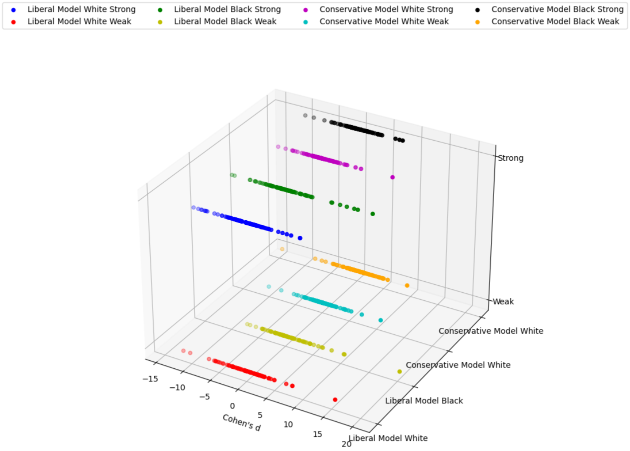
ii) Strong vs Weak w.r.t Islam vs Christianity



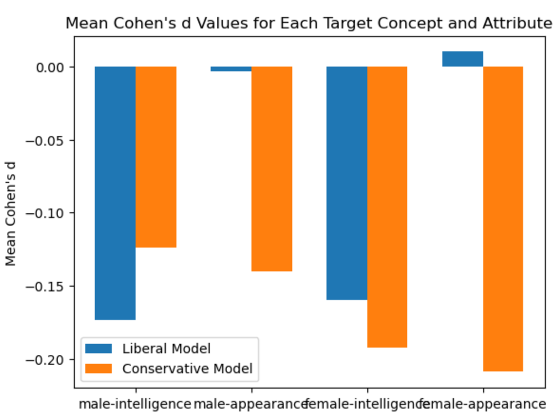


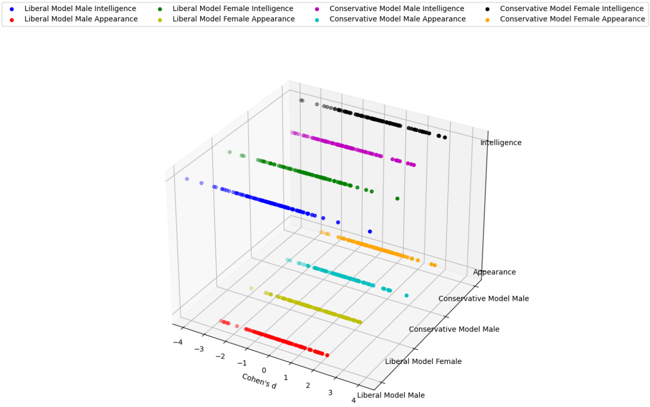
iii) Strong vs Weak w.r.t White vs Black



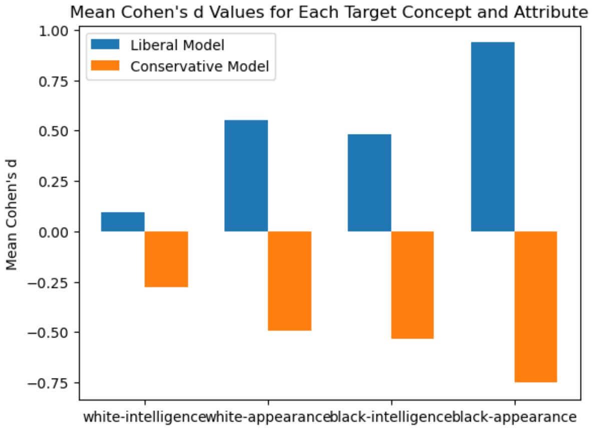


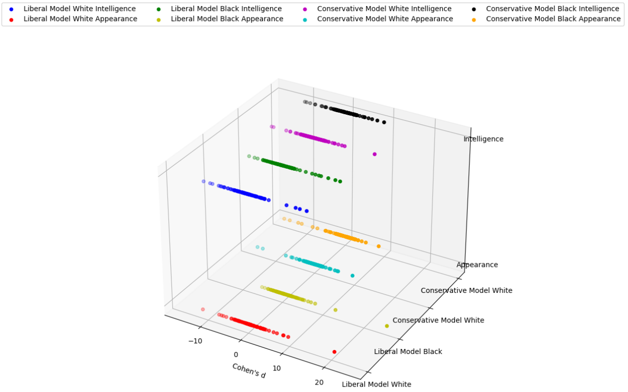
iv) Intelligence vs Appearance w.r.t Male vs Female



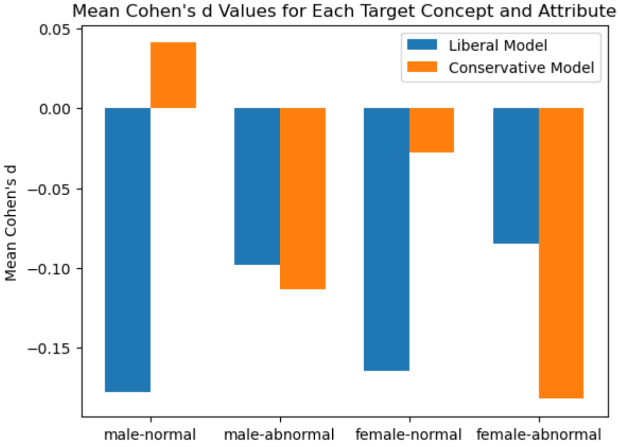


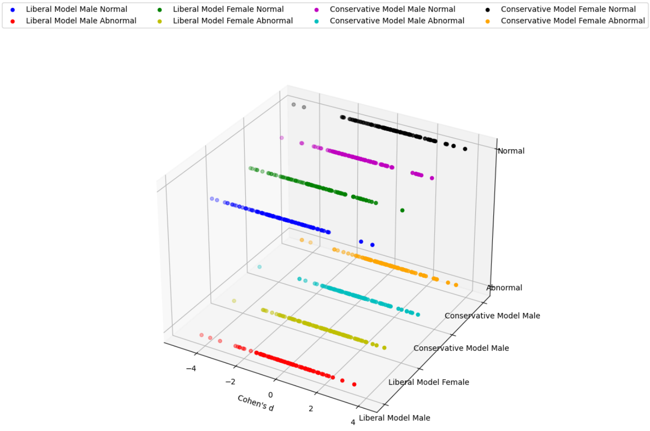
v) Intelligence vs Appearance w.r.t White vs Black



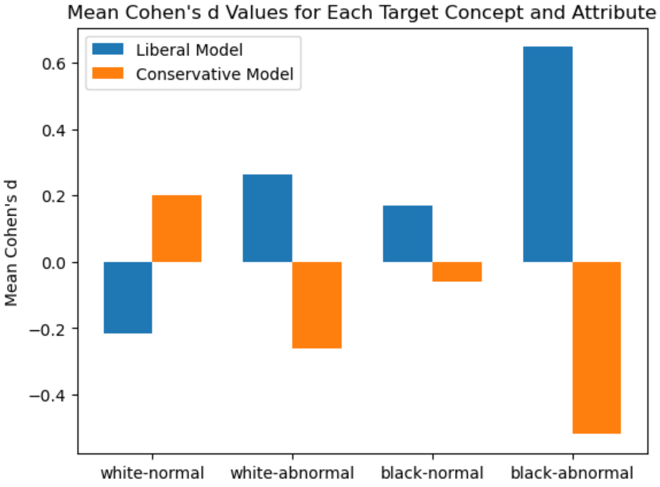


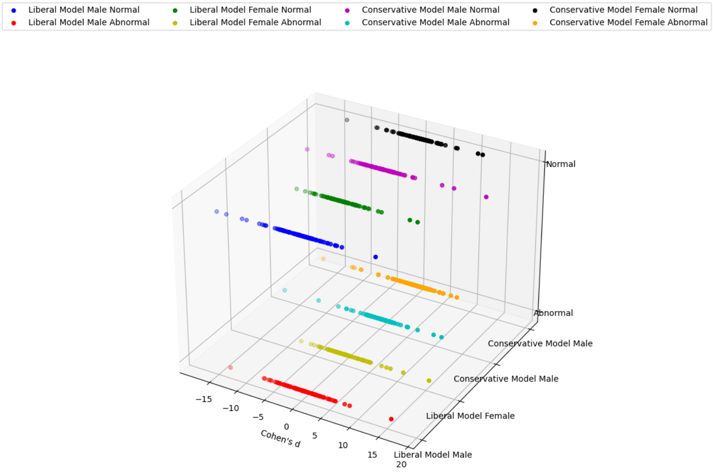
vi) Normal vs Abnormal w.r.t Male vs Female





vii) Normal vs Abnormal w.r.t White vs Black





i) For Strong vs Weak w.r.t Male vs Female Cohen's d values for both models are negative, which means that both models associate the terms "male strong," "male weak," "female strong," and "female weak" with negative qualities. However, Cohen's d values for the conservative model are slightly lower than Cohen's d values for the liberal model. This suggests that the conservative model is slightly less biased against both men and women.

- Both the liberal and conservative models are slightly biased against both men and women.
- The conservative model is slightly less biased against men than the liberal model.

- The liberal model is slightly less biased against women than the conservative model.

ii) For Strong vs Weak w.r.t Islam vs Christianity

Cohen's d values for the conservative and liberal models suggest that the liberal model is slightly more biased against Islam than the conservative model. The liberal model has a higher Cohen's d value for Islam strong and Islam weak than the conservative model. This suggests that the liberal model is more likely to associate Islam with negative qualities than the conservative model.

- The liberal model was trained on news articles from liberal sources, which may be more likely to report on negative stories about Islam.
- The liberal model may be more likely to be used by people who have negative views of Islam.
- The liberal model may be more likely to be used to generate text that is critical of Islam.

iii) For Strong vs Weak w.r.t White vs Black

Cohen's d values for the liberal and conservative models suggest that the liberal model is slightly more biased in favor of black people than the conservative model. The liberal model has a higher Cohen's d value for black strong and black weak than the conservative model. This suggests that the liberal model is more likely to associate black people with positive qualities than the conservative model.

- The liberal model was trained on news articles from liberal sources, which may be more likely to report on positive stories about black people.
- The liberal model may be more likely to be used by people who have positive views of black people.
- The liberal model may be more likely to be used to generate text that is positive about black people.

iv) For Intelligence vs Appearance w.r.t Male vs Female

The liberal model is slightly biased against men's intelligence and appearance, while the conservative model is slightly more biased against men's intelligence and appearance. The liberal model is slightly biased in favor of women's intelligence and appearance, while the conservative model is slightly more biased against women's intelligence and appearance.

- The liberal model is slightly more likely to associate men with negative qualities, such as being less intelligent and less attractive, than the conservative model.
- The liberal model is slightly more likely to associate women with positive qualities, such as being more intelligent and more attractive, than the conservative model.

v) For Intelligence vs Appearance w.r.t White vs Black

The liberal model is slightly biased in favor of black people and their appearance and intelligence, while the conservative model is slightly biased against black people and their appearance and intelligence.

- The liberal model was trained on news articles from liberal sources, which may be more likely to report on positive stories about black people's appearance and intelligence.
- The liberal model may be more likely to be used by people who have positive views of black people's appearance and intelligence.
- The liberal model may be more likely to be used to generate text that is positive about black people's appearance and intelligence.

vi) For Normal vs Abnormal w.r.t Male vs Female

The liberal model is slightly more biased against men, while the conservative model is slightly more biased against women. The liberal model is also slightly more biased against people with abnormal conditions, while the conservative model is slightly more biased against people with normal conditions.

- The liberal model was trained on news articles from liberal sources, which may be more likely to report on negative stories about men.
- The liberal model may be more likely to be used by people who have negative views of men.
- The liberal model may be more likely to be used to generate text that is critical of men.

vii) For Normal vs Abnormal w.r.t White vs Black

The liberal model is slightly more biased against white people with normal conditions, while the conservative model is slightly more biased against black people with normal conditions. The liberal model is also slightly more biased in favor of black people with abnormal conditions, while the conservative model is slightly more biased against black people with abnormal conditions.

- The liberal model was trained on news articles from liberal sources, which may be more likely to report on negative stories about white people with normal conditions.

- The liberal model may be more likely to be used by people who have negative views of white people with normal conditions.
- The liberal model may be more likely to be used to generate text that is critical of white people with normal conditions.

6.DISCUSSIONS:

1. Interpretation of our Results:

Word embedding models are trained on large amounts of text data, and they can reflect the biases that exist in that data. For example, a word embedding model trained on news articles from conservative sources may be biased against women and people of color. This bias can have a negative impact on the accuracy of NLP applications, such as sentiment analysis and machine translation.

It is important to be aware of the potential for bias in word embedding models, and to take steps to mitigate bias when using these models in NLP applications. Some ways to mitigate bias include using debiasing algorithms, fairness metrics, and considering the source of the training data.

It is also important to recognize that bias can exist in many other areas of NLP, such as data collection, preprocessing, and model architecture. Therefore, it is crucial to approach NLP research and development with a critical eye towards identifying and addressing bias in all aspects of the process. By doing so, we can create more fair and accurate NLP systems that better serve the needs of all users.

2. Takeaways/References from our results:

Here are some of the key takeaways from my results:

Political biases in word embedding models can be reflected in the language used in news articles and other political discourse, and these biases can have political consequences. For example, a word embedding model trained on conservative news articles may learn to associate certain social groups with negative connotations. This can lead to messaging that is less effective or persuasive to members of that group.

Debiasing algorithms and fairness metrics can be used to reduce political biases in word embedding models. However, the process of debiasing can be subjective and dependent on the goals of the application.

The use of more diverse and representative training data can help to mitigate political biases in word embedding models. This can involve including data from a wider range of political sources or perspectives, as well as seeking out data that represents underrepresented groups in the political discourse.

Overall, it is important to be aware of and address political biases in word embedding models, particularly in the context of political messaging and campaigns.

3. Future Work:

1. Conduct a more fine-grained analysis of the language used in liberal and conservative news articles, with a focus on how specific topics are discussed. This could help to identify more nuanced differences in language use and bias.
2. Expand the analysis to include news sources from other political ideologies, such as centrist or libertarian sources. This could provide a more comprehensive understanding of how political leanings impact word embedding bias.
3. Develop and test debiasing algorithms that are specifically tailored to the political domain. This could involve identifying patterns of political bias in word embedding models and developing techniques to counteract those patterns.
4. Apply the insights from this research to specific NLP applications in the political domain, such as political campaign messaging or automated fact-checking. This could help to identify areas where bias in word embedding models may be particularly problematic and guide the development of more effective and equitable NLP tools for political purposes.

Overall, there are many exciting opportunities for future work in this area, and I believe that addressing bias in word embedding models is a crucial step towards creating more fair and accurate NLP applications for a wide range of domains, including the political domain.

4. Limitations of our project :

1. Data Source: As our dataset is publicly available , one can easily modify them to their required words , while performing particular attribute tests , we faced a lot of issues while performing the WEAT Score metric.
2. Data Set: Our Corpora is mainly based on large number of liberal and conservative news articles of older dates, and also our news dataset only have 3 media sources and also all articles and authors are not politically motivated, still we found its interesting

to examine if there exists any difference between media that are publicly conceived as "liberal" or "conservative".

3. Word embedding model: The study used a popular pre-trained word embedding model, Word2Vec and Fasttext, which is trained on a large corpus of text. Additionally, we tried with many different types of word embedding models such as BERT , Glove, but we did not got our expected results , we tried building models with our liberal and conservative text files , because there is some unhashable data in our large corpora , it would require a larger data cleaning , sometimes the packages associated with those particular embedding models may not work perfectly.
4. Evaluation metric: The study used WEAT Score and Cohen's D as the evaluation metric for comparing word embeddings. While WEAT Score can be unreliable in some cases, such as when the two sets of words are very similar or when the two sets of words are very different. Similarly with Cohen's d can be difficult to interpret, especially for large effect sizes.

Overall, these limitations suggest that future research could benefit from using a more diverse set of data sources, evaluating multiple types of word embedding models, and exploring more nuanced evaluation metrics.

7.Acknowledgment:

Each of us took a different word embedding model and worked on building models and calculating WEAT scores to find the biases in the gender,religion,race,gender identity and age groups.

Durga Pravallika Kuchipudi:

- Researched to find the right dataset
- Worked on Fasttext word embedding model
- Contributed to creation of the presentations and of the final report's methodology, results section
- Explored another metrics in Fasstext i.e., cohen's d along with weat score , trained a new fassttext model , finding different encoding techniques for model training in fasttext

SOUMYA SHANIGARAPU:

- Researched various websites like kaggle and Github to find a dataset with supraja
- Worked on Fasttext word embedding model for calculating weat score with Pravalika.
- Contributed to creation of the presentations and of the final report's
- introduction,abstract,background section

Supraja Pericherla:

- Researched searched various data repositories and online sources to find a suitable dataset that met our project requirements along with Pravallika.

- As part of this project, I contributed to the development and implementation of a word2vec model to calculate WEAT scores and identify potential biases in a dataset of text. My role in the project involved training the word2vec model on a large corpus of text data, and using the model to calculate the WEAT scores for two sets of target words and two sets of attribute words.
- I also conducted data analysis to interpret the results and identify any potential biases present in the text. Through my contributions to this project, we were able to gain valuable insights into the biases present in the text and develop strategies to address these biases in future analyses.
- Contributed towards report for methods and results