

Paderborn University
Institute of Computer Science
Degree Programme Human-Computer Interaction

Evaluating Data-Driven Approaches to Improve Word Lists for Measuring Social Bias in Word Embeddings

Master's Thesis

Vinay Kaundinya Ronur Prakash
Born May 12, 1995 in Hyderabad, India

Matriculation Number 6905228

1. Referee: Jun.-Prof. Dr. Henning Wachsmuth
2. Referee: Prof. Dr. Stefan Böttcher

Submission date: March 6, 2023

Declaration

Unless otherwise indicated in the text or references, this thesis is entirely the product of my own scholarly work.

Paderborn, March 6, 2023



.....
Vinay Kaundinya Ronur Prakash

Abstract

Word embeddings trained on massive amounts of human-produced text would result in word embeddings that not only capture human semantics but also eventually end up encoding various kinds of negative societal biases and stereotypes prevalent in the human-produced text [Papakyriakopoulos et al., 2020]. Metrics such as WEAT and RNSB quantify bias in these models based on a set of word-lists, each representing target social groups and bias-conveying concepts. Benchmark studies in word embedding evaluations are often known to borrow word-lists from prior work in psychology experiments (IAT) and social sciences, or are sometimes curated by the authors after analysing the underlying dataset [Antoniak and Mimno, 2021]. However, there is no rationale behind choosing word-lists.

Our aim is to be able to create word-lists based on the underlying dataset that could better represent a gender group. Based on previous work by [Antoniak and Mimno, 2021], we identify lexical factors of word-lists that could have an influence on the bias measurement. We propose three data-driven approaches for word-lists creation to tackle each of these factors of influence. Word-lists created using the proposed methods are further used as target sets to evaluate Gender bias in three different types of word embedding models.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Problem Statement	3
1.3	Research Goals	5
1.4	Structure of the Thesis	6
2	Fundamentals	8
2.1	Stereotyping, Prejudice and Discrimination	8
2.2	Artificial Intelligence and Machine Learning	10
2.2.1	Machine Learning	11
2.2.1.1	Supervised Learning	14
2.2.1.2	Unsupervised Learning	18
2.3	Natural Language Processing	19
2.4	Social Biases in Word Embedding Algorithms	21
2.4.1	Word Embeddings	22
2.4.1.1	Word2Vec	23
2.4.1.2	GloVe	25
2.4.1.3	FastText	26
2.4.2	Quantifying Social Bias in Word Embeddings	27
2.4.2.1	Implicit Association Test (IAT)	27
2.4.2.2	WEAT	28
2.4.2.3	RNSB	29
2.5	Related Work	30
3	Approaches and Implementation	32
3.1	Data Description	33
3.1.1	The Pile	33
3.1.2	Pile Preprocessing	37
3.2	Training Word Embedding Models	38
3.3	Word-lists Influence Bias Metrics	40
3.3.1	Selection of Word-lists	43

3.4	Gender Classifier	46
3.4.1	Hyperparameter optimization	50
3.5	Frequency First	56
3.6	POS Filter	58
3.7	Semantic Word Clustering	60
4	Experiments and Evaluation	63
4.1	Frequency First	66
4.2	POS Filter	69
4.3	Semantic Word Clustering	72
4.4	Discussion	73
5	Conclusion	79
5.1	Summary	79
5.1.1	Limitations	82
5.2	Future Work	82
A	Target Sets	83
A.1	Frequency First	83
A.2	POS Filter	88
A.3	Semantic Word Clustering	93
	Bibliography	95

Chapter 1

Introduction

1.1 Motivation

Word embeddings, in their most basic form, are a set of Natural Language Processing(NLP) algorithms for mapping words into numerical vectors [Papakyriakopoulos et al., 2020]. Word embeddings are word representations(w), typically in the form of a d -dimensional vector, $\vec{w} \in R^d$ (typically in the range $50 \leq d \leq 300$). According to research [Ruede et al., 2017], word embeddings can be employed as features to improve the predictions and inferences of machine learning models. Recent research has established word embeddings as a prerequisite for a wide range of NLP downstream applications including machine translation [Qi et al., 2018], next sentence prediction [Mandelbaum and Shalev, 2016], sentiment analysis of text [Çano and Morisio, 2019], document screening tasks [Carvallo and Parra, 2019], credit-worthiness assessment [Dev et al., 2019] etc. Massive amounts of human-produced text would result in word embeddings that not only capture human semantics but also eventually end up encoding various kinds of negative societal biases and stereotypes prevalent in the human-produced text. These stereotypes and prejudices even get amplified in downstream NLP applications, relying on such biased word embeddings.

Keene [2011] describes social bias as *stereotyping, prejudice, or discrimination* directed towards an individual or a group of individuals in the society formed on the basis of gender, ethnicity or age. Stereotyping, prejudice, and discrimination are all related, but they are also distinct. According to Susan Fiske and Lindzey [1998], stereotyping is fixed, over generalized belief about a particular social group. For example, "Women are not aggressive" and "Men are athletic". Traditionally stereotyping is seen as the most cognitive component and prejudice is the affective component of social bias. Prejudice is a negative attitude and feeling toward a social group, for example, "All

Asians are of Chinese origin". Discrimination occurs when someone acts on prejudiced attitudes towards a social group and is thus seen as the behavioural component of social bias.

Measuring such human-like social biases is key for better understanding and addressing unfairness in word embeddings. This is often done via statistical measures, which quantify the difference in the output of a word-embedding model based on a selected dimension(such as gender, race etc). Roots to some of these statistical measures (bias metrics, here onwards) stem from Implicit Association Tests(IAT), tests used in psychological studies often to measure bias in people [Greenwald et al., 1998]. A typical IAT records subjects response times when asked to associate/pair two concepts. With this as the stimuli, IAT shows that smaller response times were recorded in concepts subjects perceive to be similar verses in pairs of concepts they perceive to be different. For example, one of the tests was where subjects were asked to associate African-American names and European-American names with "pleasant" and "unpleasant" words. The response times showed the presence of significant racial bias in many cases. This approach of detecting bias through differential association strength has been adapted and extended to measure bias in word embeddings by [Caliskan et al., 2016], which is formally called as the Word Embedding Association Test (WEAT).

The WEAT as a statistical measure, replaces the response times of the subjects, with cosine similarity between the word embeddings. For a given test, the WEAT takes two sets of attribute words(words which are suspected to be biased towards a social group, aka bias concepts) and two sets of target words(words that represent a certain social group and are assumed to associate with the bias concept). WEAT gives us a score(along with effect size and one-sided p-value) by measuring the differential association(using cosine similarity) of the two sets of target word embeddings(eg, career/family words, pleasant/unpleasant words) with the two attribute word embedding sets(eg, male/female words, African/European names). As a rule of thumb, the higher the absolute value of effect size larger the bias between words in the target sets with respect to the words in the attribute sets.

It is now obvious that apart from the influence of factors like chosen embedding algorithm (CBOW, Skip-Gram etc), model hyper-parameters(window size, workers etc), chosen association measure(cosine similarity), the predefined list of words(target and attribute word sets), which are used as stimuli, play a crucial role in the result of a bias metric. As identified in Spliethöver and Wachsmuth [2021], the out-of-vocabulary words may influence the value of the metric score. There could be a scenario where a model is trained on a small or specialized text, and then there is a higher probability that some words in the word list may not get represented as an embedding, and this in turn affects

the metric score. Another influence on the metric score is the bias encoded by the choice of the words in the word-list itself [Spliethöver and Wachsmuth, 2021]. Without background details (such as age, gender, race, experience etc) of the author of word-list, a fair assessment of the encoded biases in the word-list cannot be made. A biased word-list would lead to biased metric scores. Not only the WEAT, but other metrics such as the Embedding Coherence Test (ECT) [Dev et al., 2019], "Direct Bias" [Bolukbasi et al., 2016] and the Relative Negative Sentiment Bias (RNSB) [Sweeney and Najafian, 2019] also rely on one such word list. This provides a lot of scope for a comprehensive assessment of the influence of word-lists on the bias metrics scores.

1.2 Problem Statement

Pre-trained Language models, trained on large amounts of textual data, are prone to inherit patterns from the text data it is trained on [Zhao et al., 2017]. While NLP technologies and language models are becoming more and more integrated into our societies and extend great benefits when deployed properly, they also pose high risks of perpetuating social biases (implicit and explicit), some of which existed some 100 years ago, imposing unfair decisions against social groups identified by gender, ethnicity, (dis)ability, religion, or similar attributes of their members. Combating such biases requires measuring the bias encoded in an embedding model so that researchers can establish improvements, and many variants of embedding-based measurement techniques have been proposed (Bolukbasi et al. [2016], Caliskan et al. [2016], Sweeney and Najafian [2019]).

Implicit Association Test (IAT), often used to quantify bias in people, forms a basis for embedding-based approaches. These approaches for model analysis, measure how an embedding model associates a certain social group (male, female etc) to some bias-conveying concept (careers, family-related words etc) and how that differs for another group. Here each of the concepts is described by a list of words, with an aim to quantify perpetuated biases. This shows that, along with explicit factors (embedding algorithm, model hyperparameters etc) that influence these methods, word-lists become critical for such methods, as they become the basis for describing the context of a certain social group or a bias conveying concept. Metrics like the WEAT [Caliskan et al., 2016], RNSB [Sweeney and Najafian, 2019], ECT [Dev et al., 2019] are all embedding-based approaches by nature and their metric scores thus have a clear reliance on the predefined word-list.

While there is a wide range of bias measurement methods, every one of the embedding-based approaches relies on a group of words (word-list) to specify

stereotypes and represent a potential social bias. But the rationale for choosing a specific word-list is often unclear. Word lists are sometimes crowd-sourced, sometimes hand-selected by researchers and sometimes drawn from prior work in the social sciences. The impact of the word-lists is not well-understood, and many previously defined word-lists come with serious limitations (Antoniak and Mimno [2021], Ethayarajh et al. [2019]). Word-lists built by the researchers are susceptible to researchers' own implicit biases. Consequently, the words within the word list might not truly describe the bias as it exists in the text. Studies also show that using two different word lists to represent the same bias on a single database can produce almost opposite results [Antoniak and Mimno, 2021].

Instabilities can also arise from the organization of the word-list and seemingly harmless linguistic features. An imprecise definition of the target group or dimension of interest could result in a word-list that is too broad and/or even include multiple concepts. For eg, a word-list can manifest cultural stigmas, when terms such as "*mom*" and "*lady*" are listed under "*Domestic Work*" category [Fast et al., 2016]. These stigmas are harmful and can interact with other demographic features like gender or age [Rebecca M Puhl, 2009], and can accidentally inflate measurements towards certain groups.

Prior work examining word-lists has shown that the frequency, POS of words and length of word-lists can affect the resulting bias measurements [Caliskan et al., 2022]. Social science literature demonstrates that people are inclined to believe a name to be more renowned if they have seen it more frequently than if they have seen it less frequently, showing the role of frequency (Banaji and Greenwald [1995], Jacoby et al. [2004]). Applying the same principle to word embeddings, researchers have shown that degree of words to represent a social group (for e.g., men) increases, as we look at subsets of only the most frequent men-associated words in the training vocabulary [Caliskan et al., 2022]. With this intuition, word-lists consisting of words that are randomly chosen without accounting for word frequency lead to partial or sometimes skewed representation of the social groups [Wolfe and Caliskan, 2021].

Beyond the frequency of words, it is also necessary to understand what types of words are more or less associated with a certain target group. In particular, studies focus on the parts-of-speech used for men versus women [Caliskan et al., 2022]. Given the stereotypes of men as more active [Cheryan and Markus, 2020], it is possible that male-associated words are more likely to be verbs. In contrast, women are perceived as non-default and therefore mostly female-associated words will more likely be adjectives and adverbs. Thus drafting a word-list without the semantic knowledge of the words, results in a word-list with a skewed dimension of interest.

To this end, we identify the following high-level problems:

- The influence of word-lists employed by embedding-based bias measures to accurately represent specific social groups is not well understood, and many previous word lists have significant limitations.
- Word-lists that do not take into account word frequency and POS (Parts of Speech) result in a partial or skewed representation of social groups.
- Lack of a systematic framework for creating and assessing word lists.

Most recent work on bias measurement relies on a word-list to ground cultural concepts in language, however, a systematic framework for understanding the different sources of instability in word-lists that can affect bias measurements is missing. If we do not pay attention to the words in the word-list, the bias measurement methods will lack foundation and the claims they support will be left open to criticism. From the observations above, it is clear that creating word-lists is a particularly vulnerable task and requires more investigation. Thus a systematic framework not only acts as guidelines for a more robust word-list generation but also allows us to evaluate the generation process itself.

1.3 Research Goals

With the aim of combating the problems mentioned above, we identify the following goals and objectives, that we try to achieve as a part of this thesis work:

- Demonstrate the influence of word-lists on the embedding-based bias metric scores.
- Investigate the influence of linguistic and lexical features of words such as frequency, semanticity and POS, on word-lists used as stimuli in embedding-based bias measurement methods.
- With the idea of having a framework for the systematic generation of word-lists, develop methods to mitigate effects induced by each of the identified factors of influence using data-driven approaches.
- Evaluate methods for word-list generations and augmentation.

As mentioned in the previous section, the social impact of encoded bias in word embeddings is becoming more significant [Weidinger et al., 2021], which has direct implications on many aspects of our society as NLP technologies using these types of word representations gain greater adoption [Dev et al.,

2019]. Different measurement methods to measure encoded social biases such as the WEAT [Caliskan et al., 2016], the ECT [Dev et al., 2019] and RNSB [Sweeney and Najafian, 2019], have shown reliance on word-lists, besides other factors like the chosen embedding algorithm, model hyperparameters etc.

Introduced by Caliskan et al. [2016], WEAT tests the embedding space for the presence of a social bias. The WEAT takes two sets of target words(T_1, T_2), and two sets of attribute words(A_1, A_2) as input and performs a hypothesis test on the following null hypothesis: There is no difference between the two sets of target words in terms of their relative similarity to the similarity with the two sets of attribute words. The Embedding Coherence Test(ECT) [Dev et al., 2019] compares embeddings of two target sets(T_1, T_2) with embeddings from a single attribute set(A). The ECT calculates the average word embedding for two target sets, and then measures the cosine similarity of each embedding to embeddings in the attribute set. RNSB [Sweeney and Najafian, 2019] also receives two attribute sets and two or more target sets as input, for evaluating unintended biases in word embeddings. These predefined word-lists(target and attribute sets), that help in representing the dimension of interest, act as input or sometimes called 'stimulus' to these bias metrics.

Our work focuses on demonstrating the influence of predefined word-lists that are used in bias measurements. Initially, we gather predefined word-lists from multiple sources (Caliskan et al. [2016], Garg et al. [2018], Ethayarajh et al. [2019], Hoyle et al. [2019]), which were used as input in previous benchmark studies, while testing for a particular bias of interest(e.g., gender bias). Using these word-lists with the bias metrics, we demonstrate that different word-lists results in different bias metric scores, when used to test for the same social bias dimension and set of word embeddings. Relying on past work [Caliskan et al., 2022], we further look into word-lists to demonstrate the influence of different linguistic and lexical factors of words like their frequency, POS and semanticity. With a comprehensive analysis of word-list properties that influence the bias metric scores, we develop approaches to neutralize their effects while creating a word-list that is representative of the biases in the embeddings.

1.4 Structure of the Thesis

The thesis is comprised of five main chapters. This introductory chapter is followed by a description of the foundations that are fundamental for the further course of the thesis in Chapter 2. It begins with a specification of the principles used in the social sciences. Then the concept of artificial learning and the general approach of machine learning are introduced. Building on this, an

introduction to NLP and techniques such as word embeddings are discussed, leading to the description of social bias metrics and word-lists. The second chapter ends with a review of existing approaches to improve word lists for measuring social bias, in related work.

The subsequent description of the practical part of this work is divided into 4 sections. The first section in Chapter 3 describes the approach to demonstrate the influence of word-lists for measuring social bias. Each of the further sections in Chapter 3 describes approaches that were used in demonstrating the influence and neutralising the effect of word-list properties such as word-frequency, word POS and semanticity on social bias measurement. This builds mainly on the knowledge of word embedding models, classification and clustering algorithms. Chapter 4 describes experiments and evaluations performed on the developed approaches. Chapter 5 will reflect the results of the previous chapters and possible starting points for future research to conclude this thesis.

Chapter 2

Fundamentals

The following chapter will first introduce and further discuss the ideas of stereotyping, prejudice, and discrimination in order to provide an understanding of the issue of social bias that plague society today. Following that, fundamental knowledge of artificial intelligence and the use of machine learning algorithms in AI, with a focus on classification and clustering approaches, is established. Based on the machine learning foundations, the field of Natural Language Processing (NLP) is covered, explaining Word Embedding models and other methods relevant to this work. This chapter further presents the relevant research works in understanding and solving the problems of social bias.

2.1 Stereotyping, Prejudice and Discrimination

As proposed by Tajfel and Turner [1986], social identity theory suggests that individuals experience collective identity depending on their participation in a group. These groups (social groups) are generally formed on the basis of age, racial/ethnic and gender identities. Individuals' social identities lead them to differentiate themselves and other significant groups into "*us*" and "*them*". People engage in inter-group comparisons to retain good social identity, displaying a favourable bias towards individuals within their social group and discriminatory behaviour towards individuals of the outside of their group.

Based on the social identity theory, Susan Fiske and Lindzey [1998] defines bias or social bias that occurs in many forms including race, gender and ethnicity. The term **social bias** is a cognitive distortion that is often used as an umbrella term for *stereotyping* (cognitive bias), *prejudice* (emotional bias) and *discrimination* (behavioural bias). It is directed against social groups or individuals from social groups which arise on the basis of one or more social characteristics (age, race, gender etc) shared by their group members. **Stereotyping** refers primarily to the cognitive level of social biases. A stereotype is a specific view or assumption (thoughts) about people based purely

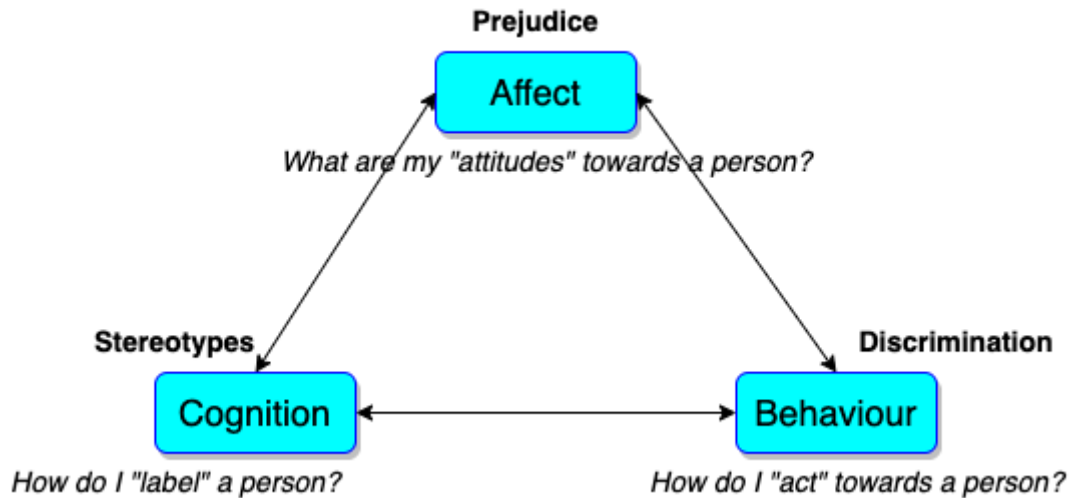


Figure 2.1: Three aspects of social bias as described by Susan Fiske and Lindzey [1998]. ABC model of bias showing *prejudice* (Affective component), *discrimination* (behavioural component) and *stereotypes* (the cognitive component). These three aspects are related, but they each can occur separately from the others.

on their group membership, regardless of their individual traits. Stereotypes can be positive or negative, and they are applied to all members of a group when they are overgeneralized. For example, the model minority stereotype of Asian Americans as highly intelligent, diligent, and mathematical can be professionally and academically detrimental [Trytten et al., 2012]. These beliefs are applied to all members of the group, despite the fact that many of the individual group members may be struggling academically and professionally.

Another well-known stereotype is perceptions of ethnic differences among athletes. According to Hodge et al. [2008], black male athletes are frequently thought to be more athletic but less intelligent than their white male counterparts. Despite a number of high-profile cases to the contrary, these ideas persist. Unfortunately, such views frequently influence how these athletes are regarded by others, as well as how they view themselves and their own skills. Stereotypes exist everywhere. Whether one agrees or disagrees with a stereotype, its content is often well-known within a specific community [Devine, 1989].

Prejudice as defined by Scheid et al. [2018], is a negative attitude and sentiment against an individual based only on one's membership in a specific social group. Prejudice against members of an unknown cultural group is common. Having a negative view toward persons who were not born in the United States is an example of bias. Persons with this prejudiced attitude may not know all people who were not born in the United States, but they despise

them because of their status as foreigners.

Discrimination is any unfavourable action taken against an individual because of their membership in a certain social group [Susan Fiske and Lindzey, 1998]. Discrimination occurs when someone acts on prejudiced attitudes toward a group of individuals. People frequently treat the target of prejudice poorly, as a result of holding negative beliefs (stereotypes) and negative attitudes (prejudice) towards that social group [Scheid et al., 2018]. Discrimination can affect institutions as well as social and political systems.

The consequences of social biases are adverse to both the group biasing and the recipient. However, the impact may be less harmful to the originator than to the recipient, the recipient experiences the most severe consequences [Scheid et al., 2018]. This is because they are the source of emotional distress. For one, social bias can have a negative impact on the physical and mental health of those affected. Frequent exposure to unfair treatments can cause psychological distress, depression and lower life satisfaction and happiness. Furthermore, social biases have severe consequences because they influence not just the individuals directly involved, but also can poison their social interactions. As a result, the overall aim should be to identify and reduce social bias as much as possible [Susan Fiske and Lindzey, 1998].

2.2 Artificial Intelligence and Machine Learning

The term **Artificial Intelligence** was coined by John McCarthy in the year 1955 [McCarthy et al., 1955] and defines it as "the science and engineering of making intelligent machines". In general, Artificial Intelligence is a computing concept that enables a machine to understand and solve complicated problems in the same way that humans do. It is also the name of the academic field of study which studies how to create computers and computer software that is capable of intelligent behaviour [Russell and Norvig, 2010].

For example, we execute a task, make mistakes, and learn from them. As part of its self-improvement, AI or Artificial Intelligence is expected to work on a problem, make some mistakes in solving the problem, and learn from the problem in a self-correcting manner. Let us consider the problem to be a game of chess. Every incorrect move diminishes your chances of winning the game. So, every time you lose against a friend, you strive to recall the mistakes you did and apply that information in your next chess game, and so on. Eventually, you improve, or in the case of artificial intelligence, the probability of winning or solving a problem improves significantly.

AI is a vast and growing field which also includes many subsets of research that have had various practical applications. *Computer Vision* is a subset of AI

that aims to simulate the visual perception component of human intelligence. Through a combination of hardware and software tools, machine vision aims to analyze images and provide predictive insights for human guidance. A popular application of computer vision is for autonomous vehicles, where vehicles use multiple cameras, lidar, radar, and ultrasonic sensors to process the visual field around the vehicle and thus make decisions on driving. Other major areas of AI include **Machine Learning**, **Natural Language Processing**, **Robotics and intelligent agents** and **Planning, Scheduling and Optimisation** processes.

Although many assumed that mathematically derived computation would be fair and neutral, resulting in AI machines with fairness beyond what exists in human society. Instead, concerns are rising that human stereotypes and prejudices are being reified into AI machines [Sweeney and Najafian, 2019].

2.2.1 Machine Learning

Machine learning (ML) is a discipline of artificial intelligence that provides machines with the ability to automatically learn from data and past experiences while identifying patterns to make predictions with minimal human intervention [Khanna and Awad, 2015]. Formally defined by Mitchell [1997], machine learning is the property of algorithms to learn from experience. A machine learning algorithm improves its performance on a task by evaluating its recent results using a performance measure and then refining its approach with this experience.

Consider the problem of bank credit approval as an example [Abu-Mostafa et al., 2012]. Suppose that a bank receives thousands of credit card applications every day, and it wants to automate the process of evaluating them. Here the bank uses historical records of previous customers in the construction of a successful formula for credit approval that can be used on future applicants.

In this example provided by Abu-Mostafa et al. [2012], the main components of the machine learning problem are the input \mathbf{x} (customer information that is used to make a credit decision), the unknown target function $f : \mathcal{X} \rightarrow \mathcal{Y}$ (ideal formula for credit approval), where \mathcal{X} is the input space (set of all possible inputs \mathbf{x}), and \mathcal{Y} is the output space (set of all possible outputs, in this case just a yes/no decision). There is a data set \mathcal{D} of input-output examples $(x_1, \mathbf{y}_1), \dots, (\mathbf{x}_N, \mathbf{y}_N)$, where $\mathbf{y}_n = f(\mathbf{x}_n)$ for $n = 1, \dots, N$ (inputs corresponding to previous customers and the correct credit decision for them in hindsight). The examples are often referred to as data points. Finally, there is the learning algorithm that uses the data set \mathcal{D} to pick a formula $g : \mathcal{X} \rightarrow \mathcal{Y}$ that approximates f . The algorithm chooses g from a set of candidate formulas under consideration, which is called the hypothesis set \mathcal{H} . For instance, \mathcal{H} could be

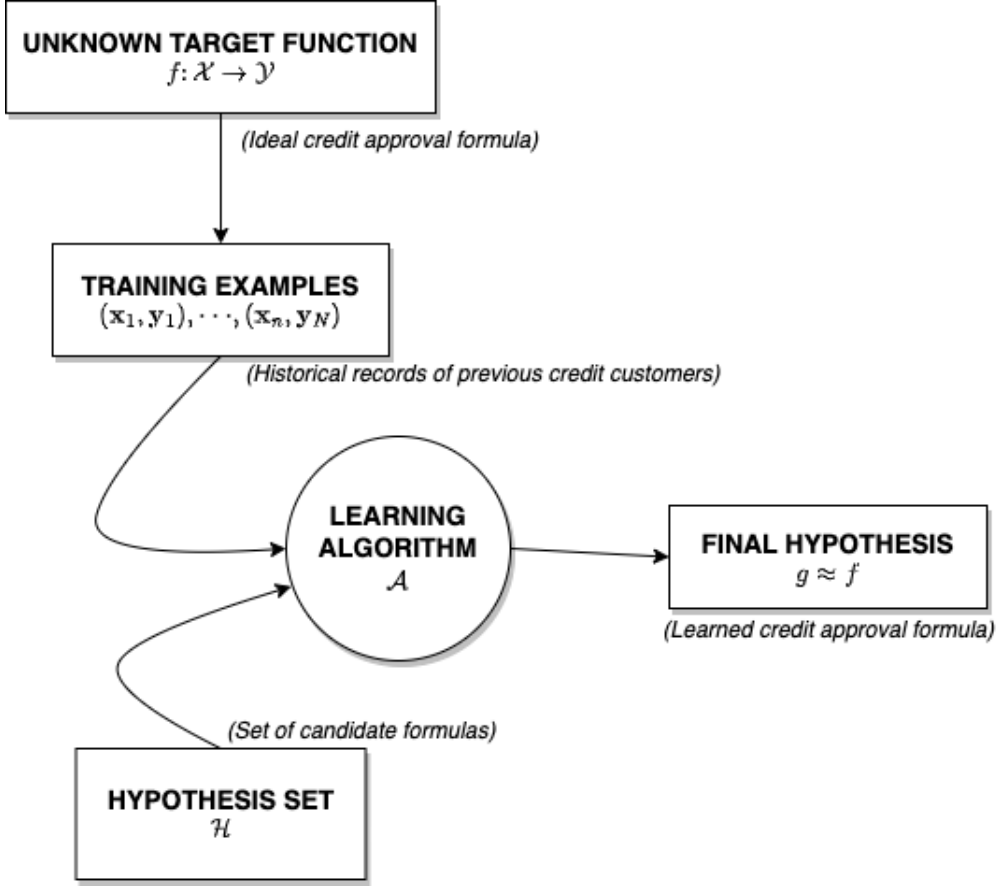


Figure 2.2: Components of a Machine Learning problem [Abu-Mostafa et al., 2012].

the set of all linear formulas from which the algorithm would choose the best linear fit to the data.

When a new customer applies for credit, the bank will base its decision on g (the hypothesis that the learning algorithm produced), not on f (the ideal target function which remains unknown). The decision will be good only to the extent that g faithfully replicates f . To achieve that, the algorithm chooses g that best matches f on the examples of previous customers, with the hope that it will continue to match f on new customers. A pictorial representation of the machine learning problem is illustrated in Figure 2.2.

Before starting the process of training a machine learning model, the given input dataset should be randomly split into three sets [Hastie et al., 2009] (2.3). The *training* set is used to train the model by adapting its parameters to the training records. Then the *validation* set is used to evaluate the trained model and optimize the training process towards the performance on the validation set. After this iterative process, the *test* set is used for the final evaluation of

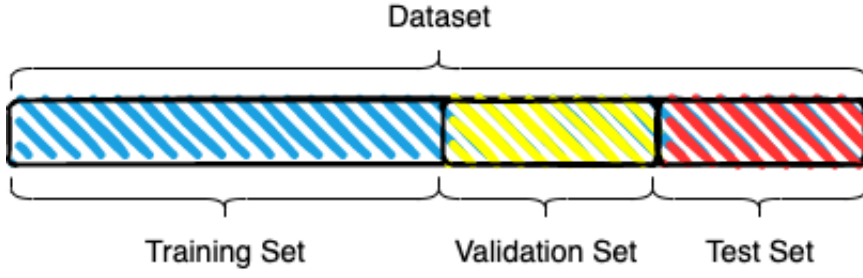


Figure 2.3: Regardless of the machine learning algorithm, the input dataset is split into training, validation and test sets.

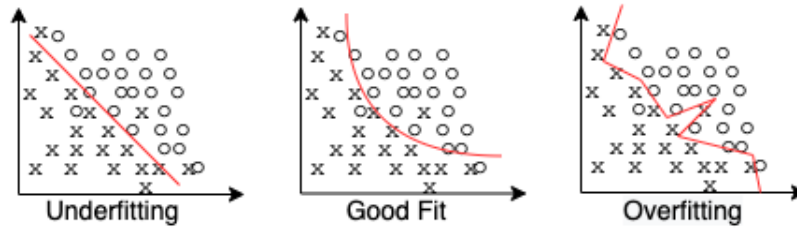


Figure 2.4: Predictions of three different models for a training dataset. The linear model (left) underfits the training data, the model in the middle has a good fit (middle), and one on the right is overfitted to the training data peculiarities.

the model on unseen data to measure its generalization capability.

Based on the generalization capability of a machine learning model, we determine if a machine learning model has a good fit [Mitchell, 1997]. *Overfitting* occurs when a machine learning model performs better than any other alternative machine learning model on the training dataset but performs worse on the testing dataset. Here the true performance (e.g., a measure of correctly classified words) of the machine learning model on unseen data is worse than necessary and thus model does not generalize well. A machine learning model is said to have *Underfitting* when it cannot capture the underlying trend of the data, i.e., it only performs well on training data but performs poorly on testing data. Ideally, the case when the model makes the predictions with 0 error, is said to have a *good* fit on the data. This situation is achievable at a spot between overfitting and underfitting, as shown in Figure 2.4.

The basic idea of machine learning is nothing but the use of a set of input data to uncover an underlying process. It is a very broad premise, and difficult to fit into a single algorithm or framework. As a result, different machine learning algorithms have arisen to deal with different situations and different

mappings between input data and anticipated output. This leads us to some of the important learning techniques, which are explained in the following sections.

2.2.1.1 Supervised Learning

In *supervised learning* setting, the input data contains explicit examples of what the correct output label should be for given input data. Supervised learning is a learning mechanism that infers the underlying relationship between the observed data (input data, \mathcal{X}) and a target variable (class label, \mathcal{Y}) [Khanna and Awad, 2015]. Learning problems with discrete target variables are referred to as *classification* problems; and *regression* problems, if the target variable is continuous. Each input data point is represented as a vector of features and these feature vectors influence the direction and magnitude of change in order to improve the overall performance of the learnt algorithm. A learnt algorithm or a well-trained model based on supervised learning algorithms can accurately predict the class labels for hidden phenomena embedded in unobserved or unknown data instances. The goal of learning algorithms is to minimize the error for a given set of input data. However, poor-quality input data (data with an imbalance in labelled examples) may encounter the problem of overfitting, which typically results in poor generalization and erroneous classification.

A reasonable example of a supervised learning problem is the optical character recognition problem [Abu-Mostafa et al., 2012]. Here the input data is a collection of images of hand-written digits, and for each image, a label of what the digit actually is. We thus have a set of examples of the form (image, digit). The learning is supervised in the sense that some 'supervisor' examines each input, in this case, an image, and determines the correct output, one of the ten categories $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$.

Classification is a type of supervised machine learning task and is considered central to developing predictive analytics capable of replicating human decision-making [Khanna and Awad, 2015]. Classification algorithms work well for problems with well-defined boundaries in which inputs follow a specific set of attributes and in which the output is categorical. The aim is to assign an input instance to the most likely class of a set of two (binary classification) or more predefined classes (multi-class classification). Let us take a closer look at a few classification algorithms that will be employed in this thesis.

Support Vector Machines A support vector machine (SVM) is a machine learning algorithm that uses supervised learning to solve complex classification, regression, and outlier detection problems by performing optimal data

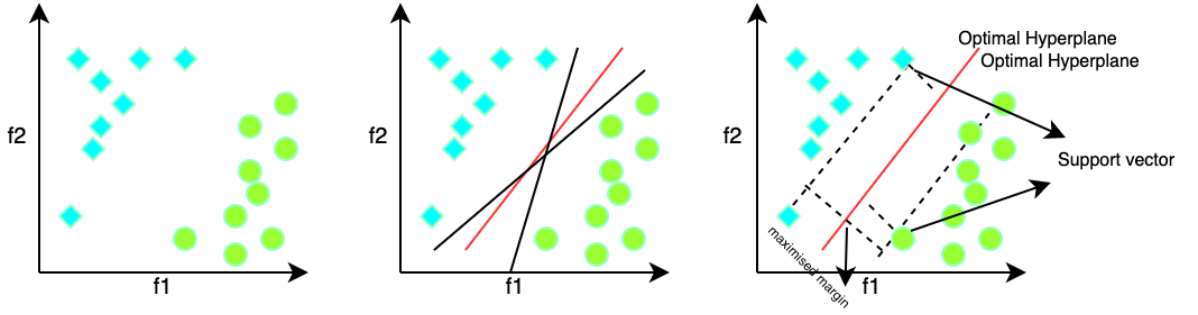


Figure 2.5: The SVM algorithm finds the hyperplane that maximizes the largest minimum distance between the support vectors for linearly separable data, [Abu-Mostafa et al., 2012].

transformations that determine boundaries between data points based on pre-defined classes, labels, or outputs. SVMs are widely adopted across disciplines such as healthcare, natural language processing, signal processing applications, and speech image recognition fields [Mitchell, 1997].

Given a set of training data in a two-class learning task, an SVM training algorithm constructs a model or classification function that assigns new observations to one of the two classes on either side of a hyperplane, making it a non-probabilistic binary linear classifier [Khanna and Awad, 2015]. The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n -dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a *hyperplane*. The data points or vectors that are the closest to the hyperplane and which affect the position of the hyperplane are termed as support vectors. The distance between the vectors (data points) and the hyperplane is called as margin. And the hyperplane with the maximum margin is called the optimal hyperplane.

The working of the SVM algorithm can be understood by using an example, shown in Figure 2.5. Suppose we have a dataset that has two tags (green and blue), and the dataset has two features f_1 and f_2 . We want a classifier that can classify the pair of coordinates (f_1, f_2) in either green or blue. As seen in Figure 2.5, it is a 2-dimensional space and by just using a straight line, we can easily separate these two classes. However, there can be multiple lines that can separate these classes. SVM here chooses the optimal hyperplane by maximizing the margin between the two distinct classes. With every new data point to be classified, the chosen hyperplane acts as the classifier.

If a dataset cannot be classified using a straight line, then such data is termed as non-linear data and the classifier used is called as *Non-linear* clas-

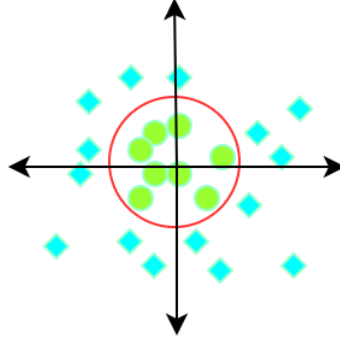


Figure 2.6: The SVM algorithm finds the optimal hyperplane using the *kernel* trick for non-linearly separable data, [Abu-Mostafa et al., 2012].

sifier (or Non-linear SVM). In such cases, the data cannot be classified using a straight line, as shown in Figure 2.6. When a problem is not linearly separable in input space, SVM cannot find a robust separating hyperplane that minimizes the number of misclassified data points and that generalizes well [Khanna and Awad, 2015]. For that, a *kernel* can be used to transform the data to a higher-dimensional space, referred to as kernel space, where data will be linearly separable. In the kernel space, a linear hyperplane can thus be obtained to separate the different classes involved in the classification task instead of solving a high-order separating hyperplane in the input space. This is commonly referred to as the "*kernel trick*" [Abu-Mostafa et al., 2012].

Classification Evaluation A classification task aims to assign an input instance to the most likely class of a set of two or more predefined classes (binary or multi-class classification). Several metrics are used for evaluating different aspects of classification. Such an evaluation is usually carried out on a test dataset reserved exclusively for testing. The test dataset has the input instances and their actual target values that act as the ground truth. The trained machine learning model predicts the target value.

For binary classification problems, the evaluation of optimal solution during the classification training can be defined based on **confusion matrix** as shown in Figure 2.7. The row of the table represents the predicted class, while the column represents the actual class. From the confusion matrix(2.7), we consider the two classes as '*positive*' and '*negative*' classes.

- **True Positive** or TP denotes the number of correctly predicted values of the positive class. Actual class = p; Predicted class = p;

		Prediction outcome		
		p	n	total
Actual value	p'	True Positive	False Negative	P'
	n'	False Positive	True Negative	N'
total		P	N	

Figure 2.7: Illustration of confusion matrix structure for binary classification. Source: [Khanna and Awad, 2015]

- **True Negative** or TN denotes the number of correctly predicted values of the negative class. Actual class = n; Predicted class = n;
- **False Positive** or FP denotes the number of wrongly predicted values of the positive class. Actual class = n; Predicted class = p;
- **False Negative** or FN denotes the number of wrongly predicted values of the negative class. Actual class = p; Predicted class = n;

The most straightforward way to measure a classifier's performance is using the **Accuracy** metric. Here, we compare the actual and predicted class of each data point, and each match counts for one correct prediction. Accuracy is then given as the number of correct predictions divided by the total number of predictions (2.1).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.1)$$

Precision is the ability of a classification model to identify only the relevant data points. Precision is the fraction of instances predicted correctly amongst all the positive predictions made (2.2).

$$Precision = \frac{TP}{TP + FP} \quad (2.2)$$

Recall is the ability of a model to find all the relevant cases within a data set. The recall is given by the fraction of positive instances correctly predicted as positive amongst all the positive instances (2.3).

$$Recall = \frac{TP}{TP + FN} \quad (2.3)$$

F_1 **Score** is computed by taking the harmonic mean of precision and recall.

$$F_1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (2.4)$$

Since multiclass classification problems deal with more than two classes ($n > 2$). Thus the metric values for binary classification outputs of all classes in the multiclass classification problem are combined to calculate a multi-class classification metric [Khanna and Awad, 2015]. **Micro-averaging** and **Macro-averaging** methods are used in this case. In macro-averaging, the performance using any of the binary classification metrics is first calculated for each class and then averaged. Whereas in the case of micro-averaging, the performances are summed up to calculate the overall performance.

2.2.1.2 Unsupervised Learning

The input data does not provide any information on correct output in the case of the *unsupervised learning*. We are just given input data examples $\mathbf{x}_1, \dots, \mathbf{x}_n$. Unsupervised learning algorithms are designed to discover hidden patterns and structures in unlabeled input data, in which the desired output is unknown [Khanna and Awad, 2015]. Two popular examples of unsupervised learning are *clustering* and *dimensionality reduction*.

For instance, if our task is to categorize a set of books into topics, and we only use general properties as features of the various books, we can identify books that have similar properties and put them together in one category, without naming that category. This learning algorithm has also found many uses in the areas of data compression, outlier detection etc.

k -Means Clustering is an unsupervised learning algorithm of vector quantization that partitions n data instances into k clusters. The algorithm defines k centroids, which act as prototypes for their respective clusters. Each data instance is assigned to a cluster with the nearest centroid when measured with a specific distance metric. When all of the data instances have been assigned to one of the k clusters, the phase of allocating data instances to clusters is complete. The process is repeated by recalculating centroids, based on previous allocations, and reassigning data instances to the new nearest centroid. The process continues until there is no change of centroids of any of the k clusters.

Generally, a k -means clustering algorithm classifies instances based on their features into k groups (or clusters) by minimizing the sum of squares of the

distances between the object data and the cluster centroid. For a given set of data instances, k means clustering partitions n instances into $k(\leq n)$ cluster sets so as to minimize the sum of squares, where ν is the mean of the data instances in each of the clusters.

$$\arg \min \sum_{i=1}^k \sum_{\mathbf{x}} \|\mathbf{x} - \mu_i\|^2 \quad (2.5)$$

The k-means clustering algorithm is easy to implement on large datasets. It has found many uses in areas such as market segmentation, computer vision, profiling applications and workloads, optical character recognition, and speech synthesis [Antoniak and Mimno, 2021, Khanna and Awad, 2015]. Evaluating the performance of a clustering algorithm is not as trivial as counting the number of errors or the precision and recall of a supervised classification algorithm.

Rand index is a measure of the similarity between two data clusters. Given the knowledge of the ground truth cluster assignments and our clustering algorithm assignments of the same samples, the rand index is a function that measures the similarity of the two assignments [Abu-Mostafa et al., 2012].

2.3 Natural Language Processing

Natural Language Processing (NLP) primarily provides computer-aided techniques for the machine recognition and processing of natural language. The goal is to facilitate direct communication between humans and computers on the basis of natural language, to facilitate interpersonal communication through machine translation, and also to cope with the increasing volumes of texts in natural language to be analyzed by machine. Natural Language Processing is formally defined by Jurafsky and Martin [2009] as "a branch of computer science and linguistics that mainly studies the automatic processing and analysis of the text". In conjunction with the development in computer performance and a variety of applications in varied domains, the long-established research area of NLP is expected to make decisive progress in the near future [Manning and Schütze, 1999].

NLP uses results from linguistics as well as methods and techniques of artificial intelligence, especially from the field of machine learning and the subfield of deep learning. NLP methods are broadly classified as *Rule-based* and *Statistical* NLP techniques. While Rule-based methods use hand-crafted rules for the analysis of language, Statistical NLP methods employ machine learning methods. Primarily, various rule-based methods and machine learning algorithms are used to identify and structure the information in texts. These

algorithms are usually sequentially applied to an input text in a pipeline. These datasets of texts are called *text corpora*.

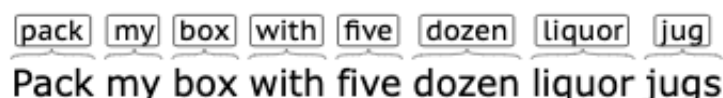
Analysis of language or application of NLP methods on text corpus is done at different levels of the language. Some of the main levels are *morphology*, *syntax*, *semantics* and *discourse*. **Morphology** focuses on the way in which words are formed, and some of the main morphological concepts are listed here:

- **Word** is the smallest unit of language that is to be uttered in isolation. Example: "men" and "swam" in "men swam".
- **Lemma** is the base form of a word as it appears in the dictionary. Example: "man" for "men", swim for "swam".
- **Wordform** is the fully inflected surface form of a lemma as it appears in a text. Example: "men" for "men", "swam" for "swam".
- **Stem** is part of a word that never changes. Example: "man" for "men", "swim" for "swam".
- **Token** is the smallest text unit in NLP: A wordform, number, symbol, or similar. Whitespaces are usually not considered tokens. Example: "men", "swam", and "." in "men swam."

While **syntax** deals with the structure of sentences and the rules for constructing them, **semantics** helps in determining the meaning of a sentence or a word in a sentence. Lastly, **discourse** deals with the analysis of texts beyond a sentence, for example, whole paragraphs or documents. With this, let us look at the following text-processing steps that are relevant to this work:

Lemmatization is the process of grouping together the different inflected forms of a word so they can be analyzed as a single item [Manning and Schütze, 1999]. Lemmatization is similar to stemming a word but it brings context to the words. So it links words with similar meanings to one word. This technique is widely used in search engines and even indexing. The lemmatization of an example sentence is shown in Figure 2.8.

Part-of-speech tagging A part-of-speech tagger maps each word in a sentence to its part of speech like nouns, verbs, adjectives and adverbs based on both its definition and its context. The Penn Treebank POS tagset distinguishes 48 different classes, like NN (noun, singular or mass), NNP (proper noun, singular), NNS (Plural Nouns) VB (verb, base form), PRP\$ (possessive



pack my box with five dozen liquor jug
Pack my box with five dozen liquor jugs

Figure 2.8: lemmatization of an example sentence. Source: <https://corenlp.run/>



VB PRP\$ NN IN CD NN NN NNS
Pack my box with five dozen liquor jugs

Figure 2.9: POS tagging of an example sentence. Source: <https://corenlp.run/>

Pronoun), VBD (verb, past tense), IN (preposition/subordinating conjunction) or CD (Cardinal Number) [Manning et al., 2014]. POS tagging for an example sentence is shown in Figure 2.9.

2.4 Social Biases in Word Embedding Algorithms

While humans find it simple to learn natural language, computers are yet to achieve this. Humans comprehend language(textual form) in a variety of ways, such as by looking it up in a dictionary or meaningfully linking it with words in the same sentence. However, computer programs are yet to be entirely human-like, requiring the use of alternative approaches to comprehend human language. Many machine learning algorithms that are used in NLP applications also require a numerical representation of a word as input. Text or word representations thus become fundamental in Natural Language Processing.

One Hot Encoding This is a type of categorical word representation and is the simplest and the most straightforward method to represent text [Naseem et al., 2021]. In this method, words are represented by either "1" or "0". In short, this method produces a vector with a length equal to the number of categories in the data set. If a data point belongs to the i^{th} category then components of this vector are assigned the value 0 except for the i^{th} component, which is assigned a value of 1. In this way, one can keep track of the categories in a numerically meaningful way.

Consider the problem of classifying a person into one of three categories: *male*, *female* and *other*. We can represent this as an array with three positions. For every person we encounter, we want to be able to represent them as a one-hot encoding in relation to our three categories. Based on this idea we can

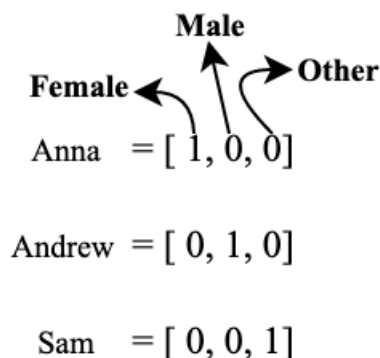


Figure 2.10: Examples of word representation using One Hot Encoding technique.

represent people as shown in figure 2.10.

However, one hot encoding and other categorical word representations (Bag of Words, TF-IDF etc) fail to capture the syntactic and semantic meaning of the words Naseem et al. [2021]. This has led to more sophisticated methods of word representations and we are going to look further into one such method called *Word Embeddings* in the following section.

2.4.1 Word Embeddings

Word Embedding is an NLP technique in which text from the corpus is mapped as vectors. A word embedding is a vector representation of a word that usually encodes the meaning of a word in a semantic or syntactic way [Jurafsky and Martin, 2009]. These vectors have many applications in downstream tasks in NLP (Search engines, Sentiment Analysis, Parsing Curriculum Vitae etc), especially in the field of distributional semantics which focuses on quantifying the meaning of linguistic items like words or tokens through their distributional properties in given dataset. In other words, it is a type of learned representation which allows the same meaning of words to have the same representation. The most significant benefit of word embedding is that it provides more efficient and expressive representation by keeping the word similarity of context and by low dimensional vectors.

A word embedding represents a word(w) as a d -dimensional word vector $\vec{w} \in R^d$. Word embedding models, trained on text corpora, aim to represent the distributional semantics of a word as a Word embedding based on the context it occurs in. With this, words with similar semantic meanings tend to have vectors that are close together and the vector differences between word

embeddings have also been shown to represent relationships between words [Bolukbasi et al., 2016]. An analogy puzzle, “man is to king as woman is to x” (denoted as $\overrightarrow{man:king}::\overrightarrow{woman:x}$), is deduced by simple arithmetic of the embedding vectors and finds that $x=\overrightarrow{queen}$ is the answer.

$$\overrightarrow{queen} \approx \overrightarrow{king} - \overrightarrow{man} + \overrightarrow{woman} \quad (2.6)$$

In some cases it can be seen that word embeddings also reflect sexism that is implicitly contained in the text corpora when a word embedding model solves the analogy “man is to computer programmer as woman is to x” with $x=\overrightarrow{homemaker}$ [Bolukbasi et al., 2016], as shown below.

$$\overrightarrow{homemaker} \approx \overrightarrow{computerprogrammer} - \overrightarrow{man} + \overrightarrow{woman} \quad (2.7)$$

Both types of word embedding models, traditional or **static** word embedding models (e.g. word2vec, Global Vectors - GloVe) and **contextual** word embedding models (e.g. Embeddings from Language Models - ELMo, Bidirectional Encoder Representations from Transformers - BERT), aim to learn a vector representation for each word in the text corpora, which are further used in downstream machine learning tasks [Garg et al., 2018].

Static word embedding models learn a global word embedding. They first build a global vocabulary using unique words in the corpora by ignoring the meaning of words in a different context. Then, similar representations are learnt for the words that appeared more frequently close to each other in the documents. The problem is that in such word representations the word’s contextual meaning (the meaning derived from the word’s surroundings), is ignored.

On the other hand, **contextual** embedding models are used to learn sequence-level semantics by considering the sequence of all words in the documents. For example, in the case of static embedding models, only one representation is learnt for “*left*” in the sentence “I *left* my phone on the *left* side of the table. However, “*left*” has two different meanings in the sentence, and thus contextual embedding models learn two different representations in the embedding space.

In our thesis, we will specifically create word embedding models with the Word2Vec, GloVe and FastText algorithms and therefore explain them in more detail in the following section.

2.4.1.1 Word2Vec

Word2vec is a word representation model developed by Mikolov et al. [2013a]. This model uses two hidden layers which are used in a shallow neural network to

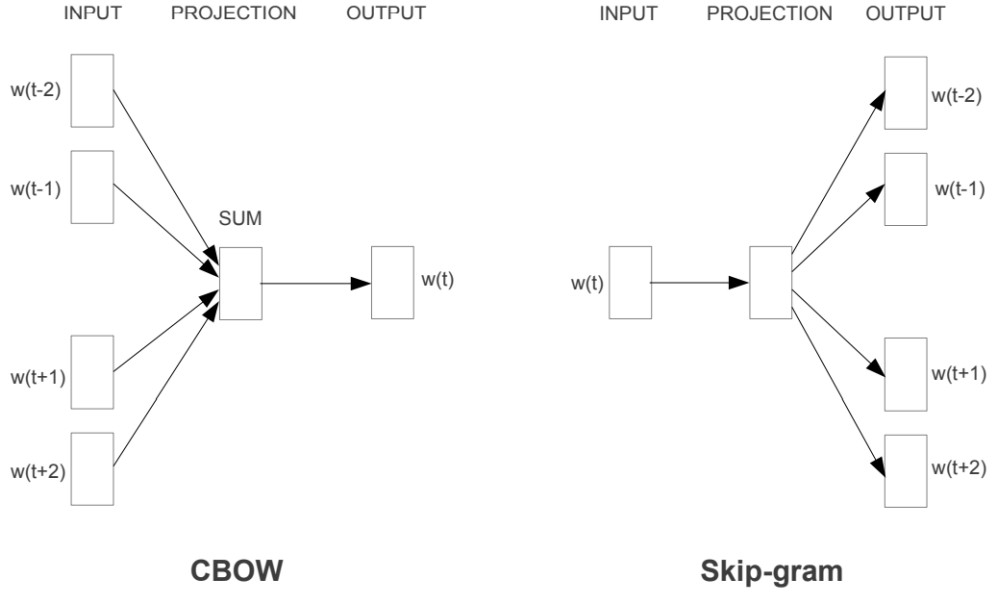


Figure 2.11: Word2Vec: CBOW architecture predicts the current word based on the context, and the Skip-gram predicts surrounding words given the current word. Source: Mikolov et al. [2013a]

create a vector of each word. The word vectors captured by Continuous Bag of words (CBOW) and Skip-gram (SG) models of word2vec are supposed to have semantic and syntactic information of words [Mikolov et al., 2013b]. To have a better representation of words, it is recommended to train the corpus with a large corpus. Word2Vec have proved to be useful in many NLP related tasks. Word2Vec was developed to build training in embedding more significantly, and since then, it has been used as a standard for developing pre-trained word representation.

Based on the context, Word2Vec predicts by using one of the two neural network models such as Continuous bag of words and Skip-gram [Mikolov et al., 2013a]. A predefined length of the window is moved together with the corpus in both models, and the training is done with words inside the window in each step. For instance, this method would regard the two words such as “small” and “smaller” near to each other in the embedding space. Figure 2.11 shows the working principle of both Word2Vec algorithms, CBOW and Skip-Gram.

Continuous Bag of Words (CBOW): The continuous bag-of-words model follows the approach of inferring the word from its given context. For each word in the sliding window, all one hot encoded vectors are input in parallel and the outputs of the input layers are averaged into one before being passed to

the hidden layer. Since the order of the individual words in the window is irrelevant due to averaging, the context in this case is modelled as a bag of words. The output vector then gives the probabilities for the words that can be inferred from the context, where the currently considered word is expected to have the highest probability [Naseem et al., 2021].

Skip-Gram: The idea behind the skip-gram model is to infer from a word the context in which it is used. A window is slid over each sentence and feeds the respective one hot encoded vectors into the neural network word by word. The output vector is a probability distribution, where the current neighbours of the word are expected to have the highest probability. The term skip-gram comes from the fact that the context of a word is modelled as an n-gram in which a single item is skipped in the original sequence [Naseem et al., 2021]. For the backpropagation, we look at the expected word neighbours in the window individually and sum up each of their errors with the output vector.

2.4.1.2 GloVe

The GloVe is a popular algorithm based on the global co-occurrence matrix, each element X_{ij} in the matrix depicts the frequency of the word w_i and the word w_j co-occur in an appropriate context window and is widely used for the text classification task [Pennington et al., 2014].

Methods like CBOW and Skip-Gram used in Word2Vec (??), are better at capturing the semantics of words but suffer significant drawbacks. Word2vec mainly focuses on the local context window knowledge and poorly utilizes the global statistical information (co-occurrence counts) [Pennington et al., 2014]. GloVe is an expansion of the word2Vec for learning word vectors efficiently where the word prediction is made based on surrounding words. The GloVe is based on the appearance of a word in the corpus, which is based on two steps. Creation of the co-occurrence matrix from the corpus is the first step, followed by factorization to get vectors as the second step.

Let the matrix of word-word co-occurrence counts be denoted by X , whose entries X_{ij} denotes the number of times word w_j occurs in the context of word w_i . Let $X_i = \sum_k X_{ik}$ be the number of times any word appears in the context of word w_i . let $P_{ij} = P(j|i) = X_{ij}/X_i$ be the probability that word w_j appear in the context of word w_i . let V be the size of the Vocabulary. Authors of Pennington et al. [2014] propose a new weighted least squares regression model and the cost function of this gives us the model (2.8)

$$J = \sum_{i,j=1}^V f(X_{ij})(w_i^T \tilde{w}_j + b_i + \tilde{b}_j - \log X_{ij})^2 \quad (2.8)$$

Here the weighting function is subjected to the following conditions,

- $f(0) = 0$, for continuous functions f .
- $f(x)$ is expected to be non-decreasing so that rare co-occurrences are not overweighted.
- $f(x)$ should be relatively small for large values of x ,

Word representation methods such as Word2vec and GloVe are simple, accurate, and can learn semantic representations of words on large data sets. They do not, however, learn embedded words from out-of-vocabulary(OOV) words. Words that are not included in the current vocabulary or words that do not appear in the current training corpus are called out-of-vocabulary words. Various models are proposed to address this challenge. We briefly describe one such model called FastText in the following section.

2.4.1.3 FastText

Introduced by Bojanowski et al. [2016], FastText is a library for efficient learning of word representations and sentence classification. FastText allows us to train supervised and unsupervised representations of words. These word embeddings can be used for numerous applications from data compression, as features into additional models, for candidate selection, or as initializers for transfer learning.

Previously seen word embedding models assign a distinct representation to every word which introduces a limitation, especially in the case of languages with sub-word level information or even in representing out-of-vocabulary words. FastText achieves good performance for word representations and sentence classification, especially in the case of out-of-vocabulary words by making use of character level information [Bojanowski et al., 2016, Naseem et al., 2021].

Each word is represented as n-grams in addition to the word itself. As an example, for the word '*matter*' and with $n = 3$, the FastText representations for the character n-grams is $\langle ma, mat, att, tte, ter, er \rangle$. \langle and \rangle are added as boundary symbols to distinguish the ngram of a word from a word itself. This helps preserve the meaning of shorter words that may show up as ngrams of other words and thus allows to capture meaning for suffixes/prefixes [Bojanowski et al., 2016]. The length of n-grams can be controlled by the $-minn$ and $-maxn$ flags for a minimum and maximum number of characters respectively. The model is considered to be a bag of words model because aside from the sliding window of n-gram selection, there is no internal structure of a word that is taken into account for featurization, i.e as long as the characters fall under the window, the order of the character n-grams does not matter.

FastText supports training continuous bag of words (CBOW) or Skip-gram models using negative sampling, softmax or hierarchical softmax loss functions. Facebook has presented pre-trained word embeddings for 294 different languages, trained on Wikipedia using FastText embedding on 300 dimensions and utilized the Word2Vec skip-gram model with its default parameters.

2.4.2 Quantifying Social Bias in Word Embeddings

Word embeddings are typically low-dimensional numerical representations of words produced by machine learning algorithms that capture word co-occurrence statistics. The assumption in these models is that words located in close proximity to one another in the embedding space are semantically similar [Mikolov et al., 2013a, Pennington et al., 2014]. For example, proximity or similarity of two words ‘*plate*’ and ‘*bowl*’, can be quantified by taking the cosine distance between the corresponding word embeddings. Word embedding models are also known to capture more analogical similarity relations. For example, by applying simple arithmetic operations to word vectors, these models capture the fact that ‘*Paris*’ is to ‘*France*’ as ‘*Rome*’ is to ‘*Italy*’ [Mikolov et al., 2013a].

However, recent research suggests that word embeddings not only encode information about mundane meanings but they also encode associations with social impact, such as ‘*woman*’ and ‘*nurse*’ or ‘*man*’ and ‘*doctor*’ [Caliskan et al., 2016, Sweeney and Najafian, 2019]. Word embeddings like these that encode subtle information on social biases, when used to solve real-world problems, end up perpetuating the learnt biases. A well-studied behavioural task developed by social psychologist Greenwald et al. [1998], to quantify social biases in human participants, is the Implicit Association Test (IAT). After the traditional methods of conducting experiments and surveys with subjects that are described in IAT, we look further into metrics used to quantify social biases in word embeddings like WEAT [Caliskan et al., 2016] and RNSB [Sweeney and Najafian, 2019].

2.4.2.1 Implicit Association Test (IAT)

The Implicit Association Test or IAT was developed by social psychologists Greenwald et al. [1998]. The IAT uses reaction time to measure participant’s associations between two target concepts (e.g. flower vs. insect) and two target attributes (e.g. pleasant vs. unpleasant) in a word categorization task. Participants are presented with a single word corresponding to one of the concepts or attributes (e.g., “rose” or “happiness”), and are asked to make a two-alternative categorization decision. In the compatible block of the test, the stereotypically associated concepts and attributes share a response key (e.g.

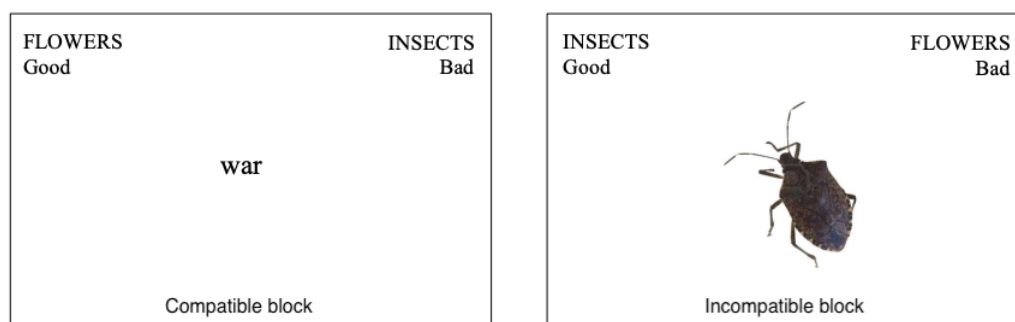


Figure 2.12: Illustration showing the compatible and incompatible blocks of a Flowers-Insects IAT. Source: Greenwald et al. [1998]

flower/pleasant vs. insects/unpleasant). In the incompatible block, the non-stereotypically associated concepts and attributes share a response key (e.g. flower/unpleasant vs. insects/pleasant). Participants tend to categorize a word more quickly in the compatible block, relative to the incompatible block, and this pattern is taken as evidence for a closer cognitive association between the compatible concept-attribute meanings, relative to the incompatible concept-attribute meanings [Greenwald et al., 1998]. For example, participants tend to respond more quickly when the response key for flower words is the same as that for pleasant words (vs. unpleasant words), suggesting that participants have a cognitive association between flowers and pleasantness [Caliskan et al., 2016].

The IAT can be used to measure the extent to which participants hold the stereotype that women are more closely associated with family life, while men are more closely associated with career life. With this ability, IAT has been conducted with a range of stereotyped concepts and attribute pairs in order to measure the strength of social biases in human participants. Metrics for quantifying social biases in word embeddings, which are based on the exact approach used in IAT are described in the following sections.

2.4.2.2 WEAT

Word Embedding Association Test (WEAT) is used to quantify bias in word embeddings. WEAT derives from the IAT (2.4.2.1) and uses word sets to represent the target social groups, along with a distance measure (cosine similarity score) between a pair of embeddings. WEAT computes the differential association scores for each word in the target concept and attribute categories [Caliskan et al., 2016].

WEAT quantifies the association of an attribute word w (e.g. 'home') with each word from the target word lists, A (e.g. 'male') and B (e.g. 'female') by

computing the association score given by the following equation 2.9.

$$s(w, A, B) = \text{mean}_{a \in A} \cos(\vec{w}, \vec{a}) - \text{mean}_{b \in B} \cos(\vec{w}, \vec{b}) \quad (2.9)$$

Effect size is computed by calculating such an association $s(w, A, B)$, for all attribute words from each of the two attribute categories. Effect size quantifies the differential association between word embeddings of two sets of target categories X, Y and two attribute sets A, B [Caliskan et al., 2016]. The mean association score for each word is then divided by the pooled standard deviation of association scores to obtain the WEAT effect size.

$$ES(X, Y, A, B) = \frac{\text{mean}_{x \in X} s(x, A, B) - \text{mean}_{y \in Y} s(y, A, B)}{\text{std} - \text{dev}_{w \in X \cup Y} s(w, A, B)} \quad (2.10)$$

The WEAT effect size takes a value between $+2$ and -2 . The WEAT metric by WEFEE [Badilla et al., 2020] also provides a *p-value* to determine if the observed WEAT *effect size* is not produced by chance. In the context of WEAT, the null hypothesis is that there is no association between the defined social groups and the bias concepts represented by the target attributes. Then a *p-value* less than an assumed significance level indicates that the observed difference in averages is statistically significant and that the null hypothesis can be rejected. For our work here, we follow the authors of Caliskan et al. [2016] and consider the bias to be statistically significant if the *p-value* is below $< 10^{-2}$.

2.4.2.3 RNSB

Relative Negative Sentiment Bias (RNSB) is based on measuring bias through word sentiment. The main idea is that if there was no bias, all words should be equally negative. Therefore, RNSB is based on calculating how negative the words in the target sets are [Sweeney and Najafian, 2019].

RNSB trains a classifier that assigns a probability to each word belonging to the negative class [Badilla et al., 2020]. Then, it generates a probability distribution with the probabilities calculated in the previous step and compares them to the uniform distribution (where all words have the same probability of being negative) using KL divergence. When the negative probability distribution is equal to the uniform one (KL divergence is 0) implies there is no bias.

Given two attribute sets A_1 and A_2 and two or more target sets (T_1, T_2, \dots, T_n) , RNSB constructs a binary classifier using set A_1 as training examples for the negative class and A_2 as training examples for the positive class. After the training process, this classifier gives every word $w \in T_1 \cup T_2 \dots \cup T_n$ a probability

that can be interpreted as the degree of association of w with respect to A_2 . RNSB then constructs a probability distribution and computes distance (using KL divergence) with a uniform distribution. The main idea behind RNSB is that the more the constructed probability distribution resembles a uniform distribution, the less biased the word embedding model is [Badilla et al., 2020, Sweeney and Najafian, 2019].

2.5 Related Work

Word embeddings and measuring biases in word embeddings is a growing area of research interest in the recent past. According to research [Ruede et al., 2017], word embeddings can be employed as features to improve the predictions and inferences of machine learning models. Recent research has established word embeddings as a prerequisite for a wide range of NLP downstream applications including machine translation [Qi et al., 2018], next sentence prediction [Mandelbaum and Shalev, 2016], sentiment analysis of text [Çano and Morisio, 2019], document screening tasks [Carvallo and Parra, 2019], credit-worthiness assessment [Dev et al., 2019] etc. But with massive amounts of human-produced text being used to train word embeddings, they not only learn all the semantic relationships in the text but eventually end up encoding various kinds of negative societal biases and stereotypes prevalent in the human-produced text.

Most works in the area of bias measurement stem from Implicit Association Tests (IAT). These are tests used in psychological studies often to measure bias in people [Greenwald et al., 1998]. Following the works of [Greenwald et al., 1998], a wide range of bias measurement methods has been proposed [Caliskan et al., 2016, Dev et al., 2019, Sweeney and Najafian, 2019]. Each of them relies on word-lists to specify stereotypes or dimensions of interest. There are also bias measurement techniques based on inference from nearest neighbours [Gonen and Goldberg, 2019]. However, there is significantly less research in the field of assessing the quality of these bias metrics. The authors of [Spliethöver and Wachsmuth, 2021], present a method to assess the quality of bias metric and the word-lists, and quantify the accuracy and robustness through Bias Silhouette Analysis. Authors of [Spliethöver and Wachsmuth, 2021] explore the aspect of word-lists influence on the result produced by a bias metric.

Authors of [Caliskan et al., 2022], analyze the dataset used in training embeddings to identify lexical factors of words which have an influence on the bias in embeddings. Authors of [Antoniak and Mimno, 2021] particularly investigated the sources of word-lists that are more often used in different bias experiments and even explored the negative influence word-lists can have on

bias measurement.

Chapter 3

Approaches and Implementation

Word embeddings, which are mathematical representations of words used in natural language processing (NLP) and other applications of artificial intelligence, can carry the social biases present in the data they were trained on [Naseem et al., 2021]. Based on the social identity theory proposed by Tajfel and Turner [1986], psychologists Susan Fiske and Lindzey [1998] define social bias as a "tendency to judge others based on group membership". Although there are many forms of social bias, including racial bias, gender bias, age bias etc, our work focuses on gender bias as it can have serious consequences for individuals and society, including limiting opportunities and perpetuating inequality [Zhao et al., 2017].

In order to debias word embeddings and improve the fairness of an NLP/AI application, the first step is to accurately measure social biases in word embeddings. Implicit Association Tests (IATs), a psychological test that is used to measure implicit/unconscious biases in human participants, form the basis for social bias metrics such as WEAT [Caliskan et al., 2016], along with RNSB [Sweeney and Najafian, 2019] and ECT [Dev et al., 2019]. These metrics (based on word-lists) are not very well evaluated and show significant limitations in precisely measuring social biases [Garg et al., 2018, Hoyle et al., 2019]. All the mentioned bias metrics rely on a word-list in order to represent a social group. Word-lists that do not take into account word frequency, POS, and semantic cohesiveness between words result in a partial or skewed representation of social groups. By identifying such factors, we propose data-driven methods to improve word-lists that are representative of intended social groups for measuring gender bias in text corpora that the word embeddings are trained on.

In this context and as part of this chapter, we aim to *demonstrate the influence of word-lists on the embedding-based bias metric scores*. We intend to further *investigate the linguistic and lexical properties of words that form word-*

lists in describing the target social groups (male and female) while measuring gender bias. With these insights, we then describe individual approaches for *systematic generation of word-lists by neutralizing the effects of each of the identified factors*.

3.1 Data Description

In this section, we describe the dataset that is used as text corpora for training the word embedding models. Then we describe the process of filtering the dataset to keep only the subsets that are relevant to our research and all the preprocessing steps applied to the text. The preprocessed text is then used for training the word embedding model.

3.1.1 The Pile

The Pile is a large (825.18 GB) English text dataset of text data that is used in natural language processing tasks, particularly for training large-scale language models, compiled by the authors of Gao et al. [2021]. It includes a variety of text data, such as books, news articles, blog posts, social media posts, youtube subtitles and more. The dataset is organized into 22 different subsets, each of which is focused on a specific type of text data. It is considered one of the important resources for researchers working on natural language processing tasks, as it provides a large and diverse collection of text data that can be used to train and evaluate large language models. In addition to its utility in training large language models, the Pile can also serve as a broad-coverage benchmark for cross-domain knowledge and generalization ability of language models [Gao et al., 2021]. A Treemap visualization of subsets of the *Pile* by their size is as shown in Figure 3.1. Short descriptions of all the subsets in the *Pile* are described in the following.

Pile-CC CC stands for Common Crawl, which is a collection of website crawls, that includes text from diverse domains of raw web pages, their meta-data and text extracts. Pile-CC which is a Common Crawl-based subset compiled by [Gao et al., 2021], uses jusText by Endrédy and Novák [2013] on Web Archive files (raw HTTP responses) for extraction, which yields higher quality output than directly using the WET files (extracted plaintext).

PubMed Central PubMed Central (PMC) is a subset of the PubMed online repository for biomedical articles compiled by the United States of America’s National Center for Biotechnology Information (NCBI). This subset consists of full-text access to nearly five million publications in Biotechnology.

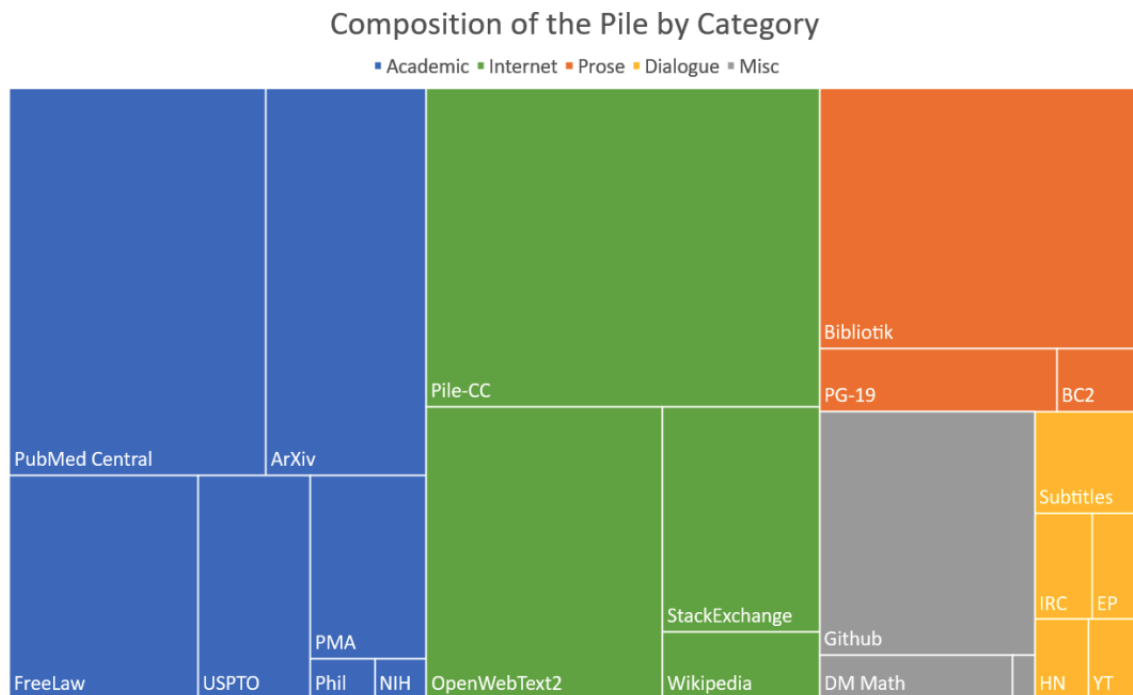


Figure 3.1: Treemap visualization of the Pile dataset and its 22 subsets. Source: Gao et al. [2021]

Books3 Books3 is a dataset of a mix of fiction and non-fiction books derived from a copy of the contents of the Bibliotik private tracker made available by Shawn Presser [Presser, 2020]. Books3 subset is invaluable for long-range context modelling research and coherent storytelling.

OpenWebText2 OpenWebText2 acts as a high quality general purpose dataset, which includes multi-lingual content from Reddit submissions until the year 2020. It also consists of the document metadata, multiple dataset versions, and open source replication code.

ArXiv ArXiv is a pre-print server for research papers that has operated since 1991 [Gao et al., 2021]. ArXiv is an open-access archive for 2,187,420 scholarly articles in the fields of physics, mathematics, computer science, quantitative biology, quantitative finance, statistics, electrical engineering and systems science, and economics.

Github GitHub is a large corpus of open-source code repositories. In order to facilitate applications like generating plausible code completions, the authors

[Gao et al., 2021] included GitHub in the hopes that it would enable better performance on code-related tasks.

FreeLaw The Free Law Project is a US-registered non-profit that provides access to academic studies in the legal realm. It consists of millions of legal opinions, legal proceedings, dockets (The official record of all documents filed by the parties or generated by the court in a case), bibliographic information on judges, and other metadata, from federal and state courts. This data is entirely within the public domain.

Stack Exchange This is an anonymized dump of all user-contributed content on the Stack Exchange network. Each site is formatted as a separate archive consisting of XML files zipped using bzip2 compression [Community, 2022]. Each site archive includes Posts, Users, Votes, Comments, PostHistory and PostLinks. It is one of the largest publicly available repositories of question-answer pairs, and covers a wide range of subjects from programming, to gardening and Buddhism.

USPTO Backgrounds USPTO Backgrounds is a dataset of background sections from patents granted by the United States Patent and Trademark Office. It contains a large volume of technical writing on applied subjects, aimed at a non-technical audience.

Wikipedia - English Wikipedia is a standard source of high-quality text. In addition to being a source of high-quality, clean English text, it is also valuable as it is written in expository prose, and spans many domains [Gao et al., 2021].

PubMed Abstracts PubMed Abstracts consist of the abstracts of 30 million publications in PubMed. PubMed is an online repository for biomedical articles run by the National Library of Medicine [Gao et al., 2021].

Gutenberg Project Gutenberg is a dataset of classic Western literature. It consists of Project Gutenberg books from before 1919 [Rae et al., 2019], which stands distinct from the more modern books in *Books3* subset.

OpenSubtitles The OpenSubtitles dataset is a dataset of subtitles from movies and television shows gathered by Tiedemann [2016]. This subset comes in handy while modelling creative writing generation tasks such as screenwriting, speech-writing and interactive storytelling.

DM Mathematics The DeepMind Mathematics dataset consists of a collection of mathematical problems from topics such as algebra, arithmetic, calculus, number theory, and probability, formatted as natural language prompts [Saxton et al., 2019].

Ubuntu IRC The Ubuntu IRC dataset is derived from the publicly available chat logs of all Ubuntu-related channels on the Freenode IRC chat server [Gao et al., 2021].

BookCorpus2 BookCorpus2 is a widely used language modelling corpus consisting of books written by “as of yet unpublished authors” [Gao et al., 2021].

EuroParl EuroParl is a multilingual parallel corpus [Koehn, 2005]. The Pile dataset uses the most current version, which consists of the proceedings of the European Parliament in 21 European languages from 1996 to 2012.

HackerNews Hacker News is a link aggregator operated by Y Combinator [Gao et al., 2021]. Users submit articles with a focus on topics in computer science and entrepreneurship. We scrape, parse, and include these comment trees since we believe they provide high-quality dialogue and debate on niche topics.

Youtube Subtitles The YouTube Subtitles subset is a parallel corpus of text compiled from human generated closed captions on YouTube. Youtube Subtitles are a source of multi-lingual, educational content, popular culture, and natural dialogue [Gao et al., 2021].

PhilPapers This subset is made up of open-access, high-quality philosophy publications from an international database maintained by the Center for Digital Philosophy at the University of Western Ontario [Gao et al., 2021].

NIH ExPorter The NIH Grant abstracts provide a bulk-data repository for awarded applications through the ExPORTER service from the year 1985 to the present.

Enron Emails Provided by the authors of Klimt and Yang [2004], the Enron Emails subset of the Pile is a commonly used dataset for research about the usage patterns of email.



Figure 3.2: Preprocessing steps for preparing the *Pile* dataset, before training the word embeddings.

3.1.2 Pile Preprocessing

Before feeding the Pile dataset into a word embedding algorithm to produce word embeddings, we perform few preprocessing steps on the dataset, as shown in Figure 3.2. The Pile, a large dataset comprising 22 subsets, is known for its diversity in data sources and comprises of text from many domains including books, GitHub repositories, webpages, chat logs, and medical, physics, math, computer science, and philosophy papers [Gao et al., 2021]. In the context of this work, we focus on word embeddings to be trained only on non-code, non-formulaic, English-language text. Thus dataset selection procedure, followed by filtering texts based on language and finally tokenization of text is performed before training word embeddings.

Subset Selection Initially we perform a *Subset Selection* procedure to keep only relevant datasets. Here we filter out code snippets in GitHub dataset and collection of mathematical problems and formulae from topics such as algebra, arithmetic, calculus etc in the DM Mathematics dataset [Gao et al., 2021].

Language Filtering Although the Pile is compiled to be a mostly English language dataset, some of its subsets like EuroParl, Youtube Subtitles, OpenWebText2, Books3, BooksCorpus2 and Gutenberg datasets consist of multi-lingual texts. With a focus to keep only English texts, we make use of a language detection library to detect the language of a text. We use the `langdetect` library made available by Danilák [2021]. Langdetect provides support for the detection of over 50 languages. Langdetect is a non-deterministic algorithm, which means that if you try to run it on a text which is either too short or too ambiguous, the algorithm produces different results every time it is run. To enforce consistent results from the algorithm, `DetectorFactory.seed` value is set to 0.

Tokenization Furthermore all the word embedding models require the texts to be tokenized. In order to achieve the same, we leverage a preprocessing method provided by `gensim.utils` called `simple_preprocess`, [Řehůřek and

Sojka, 2021b]. This method is responsible for tokenizing the text and lower-casing the tokens. By default the method ignores tokens with length less than 2 [Řehůřek and Sojka, 2021b].

Authors of the Pile Gao et al. [2021] and other recent works [] have shown that especially for large models, diversity in data sources improves general cross-domain knowledge of the model, as well as downstream generalization capability. Thus we perform the aforementioned preprocessing steps and use the *Preprocessed Pile* dataset in further training 3 word embedding models (Word2Vec, GloVe and FastText), whose details are provided in the following section.

3.2 Training Word Embedding Models

To investigate the development of social biases in word embeddings, especially in the form of Gender bias, we aim to train word embeddings using a different static word embedding algorithms. We train 3 popular word embedding models such as Word2Vec [Mikolov et al., 2013a], FastText [Bojanowski et al., 2016] and GloVe [Pennington et al., 2014] on the preprocessed Pile dataset, and their parameter settings are described below. A summary of the hyperparameter values for all 3 models is shown in the table 3.1.

Word2Vec As described in section 2.4.1.1, word2vec uses large amounts of unannotated plain text and learns relationships between words automatically. We follow the word2vec implementation of the Gensim library provided by the authors Řehůřek and Sojka [2021c]. We train embeddings using the Continuous-Bag-of-Words (CBOW) algorithm. The authors of [Pennington et al., 2014] showed that the accuracy of word vectors on an analogy test stagnates after 300 dimensions, thus the `vector_size` (Dimensionality or size of the word embedding) is set to 300.

Major training parameters that affect the speed and quality of the training process include, `min_count` (Ignores all words with total frequency lower than this value) which is set to 5 and `window` (Maximum distance between the current and predicted word within a sentence) is set to 5. To speed up the process of training, we set the value of `workers` (number of worker threads to train the model) parameter to 32. All the other parameters introduced during training are set to default values.

FastText As described in section 2.4.1.3, fastText generates word embeddings by treating each word as the aggregation of its subwords. This algorithm is especially significant in case of morphologically rich languages (For

Table 3.1: List of important hyperparameters tuned while training word embedding models, along with their set values.

Hyperparameters	Word2Vec	FastText	GloVe
vector size	300	300	300
min count	5	5	5
workers	32	32	32
window	5	5	15
training algorithm	CBOW	CBOW	GloVe
sorted vocabulary	True	True	True
batch size	100,000	100,000	100,000
epochs	5	5	5

eg, German) in which a single word can have a large number of morphological forms [Bojanowski et al., 2016]. Such morphological forms of words are not considered by Word2Vec algorithm and thus fastText does significantly better on syntactic tasks as compared to the original Word2Vec, especially when the size of the training corpus is small [Bojanowski et al., 2016].

Hyperparameters for training the FastText model follows the same pattern as Word2Vec [Řehůřek and Sojka, 2021a]. The `vector_size` parameter is set to 300 and the `workers` parameter to 32. All the other parameters are similar to as described for Word2Vec model and are set to their default values. FastText model also consists of three parameters in addition to the common parameters of word2vec model, called `min_n` (minimum length of character ngrams) set to default value of 3, `max_n` (maximum length of character ngrams) set to default value of 6 and `bucket` (number of buckets used for hashing ngrams) which is set to default value of 2000000.

GloVe Global Vectors for Word Representation or GloVe, trains word representations for which the dot product of two words corresponds to the logarithm of their probability of co-occurrence, as described in section 2.4.1.2. We follow the implementation of GloVe provided by the authors of [Pennington et al., 2014]. In general, GloVe collects unigram counts of words, constructs and shuffles cooccurrence data, and trains a simple version of the model [Pennington et al., 2014].

Following the implementation of GloVe provided by the authors of [Pennington et al., 2014], the hyperparameters that influence the speed and accuracy of training includes `VOCAB_MIN_COUNT` (Ignores all words with total frequency lower than this value), `VECTOR_SIZE`, `MAX_ITER` (Number of iterations (epochs) over the corpus) and `WINDOW_SIZE` and `NUM_THREADS` (number of worker threads to train the model). The `VECTOR_SIZE` is set to 300 and

NUM_THREADS is set to 32, keeping all the other parameters set to their default values.

3.3 Word-lists Influence Bias Metrics

After training 3 different word embedding models (*Word2Vec*, *GloVe* and *Fast-Text*) on the processed *Pile* dataset, the next goal is to establish the influence of word-lists on embedding-based social bias metrics (*WEAT* and *RNSB*). In this context, we study how the word-lists are used to represent intended social groups (e.g, Gender based groups) and target bias concepts (e.g, Careers associated with "male" and "female" groups) from Implicit Association Tests (IAT, as described in section 2.4.2.1) conducted on human participants in the past [Nosek et al., 2002a]. We then perform these tests on generated word embeddings for the *Pile* dataset using *WEAT* (Word Embedding Association Test, [Caliskan et al., 2016]) and *RNSB* (Relative Negative Sentiment Bias [Sweeney and Najafian, 2019]). We finally identify some of the word-list factors that could influence these social bias metrics and introduce data-driven approaches to improve these word-lists.

We consider 2 social bias metrics that adapts from the Implicit Association Test, Word Embedding Association Test [Caliskan et al., 2016] and Relative Negative Sentiment Bias [Sweeney and Najafian, 2019] for measuring human-like social biases present in word embedding models. IAT as proposed by the authors Greenwald et al. [1998], is used to document social biases in humans, persisting in the form of stereotypes and prejudices (section 2.1). As seen previously, IAT quantifies the social bias by measuring the difference in response times of human participants, when asked to associate two concepts they find similar and two concepts they find different [Greenwald et al., 1998].

Before we look into some of the popular Gender based IAT studies conducted in the past [Caliskan et al., 2016, Nosek et al., 2002a], we define "*stimuli*" used in the IATs. In Implicit Association Tests, stimuli refer to the materials presented to participants during the test and the stimuli could include words, pictures, or other materials that are used to measure implicit biases [Greenwald et al., 1998]. The stimuli used in the IAT are chosen based on their association with specific social groups and target bias concepts, such as gender, race, and ethnicity associations [Greenwald et al., 1998]. The selection of stimuli in an IAT becomes a critical aspect of the test [Greenwald et al., 1998], as the words used in the test must accurately represent the intended social groups and target bias concepts in order for the results of the test to be meaningful and accurate.

Career-Gender IAT		
Social Groups	male	<i>John, Paul, Mike, Kevin, Steve, Greg, Jeff, Bill</i>
	female	<i>Amy, Joan, Lisa, Sarah, Diana, Kate, Ann, Donna</i>
Bias Concepts	career	<i>executive, management, professional, corporation, salary, office, business, career</i>
	family	<i>home, parents, children, family, cousins, marriage, wedding, relatives</i>

Table 3.2: Word-lists used to represent the social groups and bias concepts and presented as stimuli to participants in the *Career-Gender* IAT conducted by the authors Nosek et al. [2002a].

Study 1 The *Career-Gender* Implicit Association Test (IAT) performed by Nosek et al. [2002a] is a study aimed at measuring implicit biases related to gender and career associations. For measuring occupation related gender biases (stereotypes and prejudices) that participants might have about traditional gender roles and whether one unconsciously assigns certain roles to women or men, a *Career-Gender IAT* was proposed by Nosek et al. [2002a]. Career-Gender IAT presents human participants with word-lists representing the target *social groups* (Gender in this case, one for each binary gender group *male* and *female*) and word-lists representing the target bias concept (list of *career* related terms associated with men and list of careers or domestic role terms (*family*) associated with women). Stimuli or the word-lists used in the career-gender IAT conducted by the authors Nosek et al. [2002a] is shown in the table 3.2.

From the social group word-lists, participants are then asked to associate each of the words to "career" or "family" terms as quickly as possible [Greenwald et al., 1998, Nosek et al., 2002a]. The results of the Career-Gender IAT have shown that participants have faster reaction times when categorizing words that are consistent with gender stereotypes. In total, 38,797 human participants fully completed the test and the authors Nosek et al. [2002a] confirmed that female names were found to be more associated with *family* than *career* related words, with a reported effect size of 0.72 and p-value $< 10^{-2}$

Study 2 A gender-based Implicit Association Test (IAT) for math versus arts measures implicit biases related to gender and academic domain associations [Nosek et al., 2002b]. This *Math-Arts* IAT was designed to measure the implicit biases that people may have towards men and women in different academic domains, specifically in "math" and "arts" [Nosek et al., 2002b].

The stimuli used in this IAT typically include words related to math and words related to arts, as well as male and female terms, as shown in table 3.3.

Math-Arts IAT		
Social Groups	male	<i>male, man, boy, brother, he, him, his, son</i>
	female	<i>female, woman, girl, sister, she, her, hers, daughter</i>
Bias Concepts	math	<i>math, algebra, geometry, calculus, equations, computation, numbers, addition</i>
	arts	<i>poetry, art, dance, literature, novel, symphony, drama, sculpture</i>

Table 3.3: Word-lists used to represent the social groups and bias concepts and presented as stimuli to participants in *Math-Arts* IAT conducted by the authors Nosek et al. [2002b].

As per the IAT, participants are asked to categorize these stimuli into different categories, such as "male" and "female" or "math" and "arts". For example, if a participant has faster reaction times when categorizing math words with male faces and arts words with female faces, this may indicate an implicit bias towards men in math and women in arts and vice-versa [Nosek et al., 2002b]. Authors conducted *Math-Arts* IAT on 28,108 human participants and reported an effect size of 0.82 and p-value $< 10^{-2}$, indicating that "female" terms were more associated with "arts" than "math" related domains.

The Implicit Association Test is a well-established tool for measuring implicit biases in social cognition, specifically in human participants [Greenwald et al., 1998]. However, the traditional IAT has limitations in terms of its ability to measure complex biases and its generalizability to diverse groups and cultures [Nosek et al., 2002a]. To overcome these limitations, recent studies have developed embedding-based metrics, such as the Word Embedding Association Test (WEAT) [Caliskan et al., 2016] and the Relative Negative Sentiment Bias (RNSB) metric [Sweeney and Najafian, 2019].

As described in the section 2.4.2.2, WEAT adapts the IAT by using word embeddings, to measure implicit biases for various social groups (categorized by gender, race etc) by comparing the semantic similarity between word embeddings [Caliskan et al., 2016]. Similar to IAT, WEAT defines *Target set* and *Attribute set* in order to represent a particular social category (e.g., male or female names) and specific bias concepts (e.g., career, math or arts). Thus these word-lists are then considered analogous to "stimuli" that is used in IAT.

Target set (denoted by T) corresponds to a set of words intended to denote a particular social group, which is defined by a certain criterion. For example, if the criterion is gender we can use it to distinguish two groups, women and men. Then a Target set representing the social group "women" could contain words like "she", "woman", "girl" etc.

Attribute set (denoted by A) is a set of words representing some attitude, characteristic, trait, occupational field, etc. that can be associated with individuals from any social group. For example, the set of "art" attribute words could have words like "poetry", "dance", "literature".

Similarly, RNSB (described in section 2.4.2.3), adapts the IAT by using word embeddings and measures implicit biases in social cognition by comparing the proximity of words in semantic space [Sweeney and Najafian, 2019]. RNSB defines target and attribute sets in a similar way as the WEAT, but also considers the distribution of words in semantic space [Sweeney and Najafian, 2019]. Henceforth we refer to any word-list representing a social group as *Target set* and word-list representing a biased concept as *Attribute set*.

Target and Attribute set for evaluating word embeddings on different bias aspects, are often borrowed from state-of-the-art IAT experiments and other prior work in psychology and social sciences [Antoniak and Mimno, 2021, Greenwald et al., 1998]. Since the selection of word lists (target and attribute sets) is primary to bias metrics and bias evaluation in word embeddings, we discuss in the following section some of the word-list selection methods used by researchers in the past.

3.3.1 Selection of Word-lists

Manually inspecting the terms in the target sets used in two examples IAT studies (section 3.3 and 3.3), we see that the terms used in the two target sets are totally different. While targeting sets in study 1 consist of only proper nouns (names, e.g., *Amy, Joan, Sarah, ... John, Paul, Kevin* etc) to represent "male" and "female" social groups, target sets in study 2 comprises of pronouns and common nouns (e.g., *she, her, girl, .. he, him, his* etc) in representing the same social groups. While different target sets could lead to different bias evaluations, the authors of Sedoc and Ungar [2019] demonstrate that different classes of words (e.g., names vs. pronouns) can sometimes represent an unintended dimension (e.g., age instead of gender) of the social group. However, the rationale for choosing terms in the target sets is not explained by the researchers [Greenwald et al., 1998, Nosek et al., 2002b]. Based on the work by authors Antoniak and Mimno [2021], we describe some of the sources of target sets leveraged in some of the benchmark word embedding bias evaluations.

- **Borrowed from Literature:** Benchmark studies in word embedding evaluations are often known to borrow word lists from prior work in psychology experiments(IAT) and social sciences [Antoniak and Mimno, 2021]. Authors of the WEAT Caliskan et al. [2016], choose word-lists

from the Implicit Association Tests [Greenwald et al., 1998] to validate the previously performed gender based bias experiments. Authors Garg et al. [2018] also borrow the list of personality traits from the adjective checklist provided by Williams and Best [1977]. The population datasets collected and compiled by Government also act as sources for word-lists [Antoniak and Mimno, 2021]. Authors Bolukbasi et al. [2016], Caliskan et al. [2016] used U.S. census data and data from the U.S. Bureau of Labor Statistics to gather names and occupations that are common to certain demographic groups.

- **Adapted from Lexical Resources:** Word lists are also drawn from existing dictionaries, lexicons and other public resources, such as SemEval tasks [Zhao et al., 2017] and ConceptNet [Fast et al., 2016].
- **Curated and Re-Used:** Word-lists are sometimes hand-selected by the authors, usually after close reading of the datasets used to generate word embeddings [Antoniak and Mimno, 2021]. Authors of Joseph et al. [2017], hand-select a set of identity terms based on their frequency in a Twitter dataset. Many experiments also directly re-use word-lists that were used in other bias measurement experiments. For example, the word-lists used in research by Bolukbasi et al. [2016], Caliskan et al. [2016] are often re-used to test gender bias on many different datasets [Ethayarajh et al., 2019].

These word-list selection methods seem to be suitable for choosing word-list terms at first, but upon close inspection, each source suffers from a number of limitations [Antoniak and Mimno, 2021, Ethayarajh et al., 2019]. While borrowing word-lists from past literature provides a good starting point in bias measurement, researchers are still responsible for examining the word-lists and verifying whether they can represent the intended social group [Antoniak and Mimno, 2021]. The Government datasets which are a good source of terms specific to certain demographic groups, tend to be information specific to one particular country or region. Thus these word-list sources are seen to be inadequate, as they tend to assume a level of representation which might not be true when used to evaluate a more universal (consisting of word-wide data) dataset [Caliskan et al., 2022]. While the word-lists sourced from lexical resources come pre-validated, these word-lists do not work well in the case of newly found domains of data [Antoniak and Mimno, 2021]. Although hand-curated and re-used word-lists can result in increased precision in word-lists, this method relies heavily on the authors’ own social biases [Antoniak and Mimno, 2021].

These limitations imply that the existing methods for sourcing word-lists are not well understood or evaluated. Authors Antoniak and Mimno [2021] also establish that the word-list limitations like varied word-list sources, authors' personal biases and even some linguistic features of the terms in the word-lists give rise to a series of instability factors that could be encoded in the word-lists, which results in inaccurate social group representations and bias metric measurement. We categorize the word-list instabilities under 3 broad groups as *definitional*, *lexical* and *word-list size* factors and discuss the implications of each of these factors in the following.

Definitional Factors: The authors of [Antoniak and Mimno, 2021] discuss the implications of reductive definitions of target concepts, the imprecise definition of target concepts and including confounding concepts in word-lists as definitional factors that might influence the word-list creation. Word-lists using names (proper nouns) to represent social groups or bias concepts like race could result in a distorted representation of the underlying dataset used to generate the word embeddings [Nguyen et al., 2014a]. Authors of [Nguyen et al., 2014a] also argue that representing a gender group with binary extremes (male and female) results in a poor representation of the social group in a large diverse dataset. Word-lists could result in a broad list, including multiple concepts if the target bias concept is not well defined [Ethayarajh et al., 2019].

The authors [Antoniak and Mimno, 2021] also point out that there has been little work in successfully connecting bias research in ML and NLP with literature in psychology and prominent race studies, as engaging with such literature would provide a better foundation for making decisions on word-list creation.

Lexical Factors: Prior work examining word-lists has shown that lexical features of words like the frequency and part of speech of individual terms used as part of the word-lists (for representing social groups) can affect the resulting bias measurements [Antoniak and Mimno, 2021]. Authors of [Ethayarajh et al., 2019] examine the frequency of words in word-lists and confirm that WEAT tests require the paired target sets to occur at similar frequencies. Apart from frequency, the authors of Caliskan et al. [2022] also investigate the parts-of-speech (POS) and meanings associated with terms representing the social group to suggest that POS of words associated with "men" and "women" are essentially different. For example, given that "men" are perceived as more active agentic than "women" [Hsu et al., 2022], it is possible that male-associated words will be more likely to be verbs and female-associated words to be adjectives and adverbs. These findings thus provide a lead to investigate the type of words and associated frequencies while creating word-lists to

correctly represent a social group.

Word-list size Factors: Size of the word-lists and the order of terms in the word-lists are also considered factors while drafting word-lists. Although authors of Kozlowski et al. [2019] confirm a small increase in bias measurement when lengthy word-lists were used, the length of the word-lists are not well investigated and is seen as a major factor in word-list creation. Prior works show that not only the size of the word-lists but the ordering of terms in one list with another word-list, could also affect the bias measurement in some cases [Antoniak and Mimno, 2021].

Having identified word-list sources and examined word-list factors that could influence the bias measurement, it is clear that word-lists which do not count for the underlying dataset are inefficient in representing the intended social groups and thus results in inaccurate bias measurement. With a focus on the identified *lexical* factors, we aim to analyse the frequency distribution of the terms used in the word-lists, investigate the type of terms or POS of terms and finally incorporate these insights in word-list creation. In this regard, we propose 3 data-driven methods to create word-lists that could improve the social group representation and bias measurement in the following sections. Using the proposed methods, we aim to create *target sets* for representing the gender groups (male and female). We then evaluate the trained word embeddings (Word2Vec, FastText and GloVe embeddings trained on the Pile dataset) by comparing the social bias metric (WEAT and RNSB) measurement values, using the newly created target sets in the two gender based IATs (section 3.3). Before looking into the exact details of the methods proposed, we describe a word gender classifier which assigns one of the three class labels (male, female or others) to the words in the underlying data. This gender classifier helps in identifying gender associated with words (word embeddings) and acts as an integral part in all the 3 methods.

3.4 Gender Classifier

The proposed methods which are described in the following sections (3.5, 3.6, 3.7), rely upon a Gender classifier. In the context of this work, we aim to identify the gender associated with each word in the trained word embedding models and then choose words from each gender group as entries into word-lists which are later used in social bias metrics for gender bias measurement. Even though the two main gender groups under consideration in this work are "female" and "male", words are also classified into a "neutral" group as more often words in English language are gender-neutral (e.g., *child*, *kid*, *member*,

Gender Classifier Dataset				
Dataset	Male	Female	Neutral	Example words
MDGender (<i>names_gender</i>)	10,807	18,686	-	<i>John, Emma..</i>
MDGender (<i>gendered_words</i>)	3,030	3,021	-	<i>father, mother..</i>
WordNet (<i>neutral_words</i>)	-	-	8,921	<i>actor, adult..</i>
Total	13,837	21,707	8,921	

Table 3.4: A summary of the datasets used in training and evaluating the *Gender classifiers*.

person etc) and wrongly classified neutral words as *female* or *male* could influence the representation of gender groups in word-lists [Caliskan et al., 2022]. In such cases a binary classifier becomes insufficient, thus here we employ a supervised multi-class classification approach, where we classify words into three classes: "female", "male", and "neutral".

Previous work on gender classification has been predominantly supervised task and relied mainly on lexicons, that are explicitly binarily gendered (e.g., he, boy, dad.. vs she, girl, mom) [Bolukbasi et al., 2016]. However, we train a classifier to classify words into 3 classes, to account for gender neutral words and not misclassify words. To achieve this we make use of a popular dataset, MDGender [Dinan et al., 2020], which is considered a gold-labeled dataset for the masculine and feminine classes and a list of gender neutral words that were derived from WordNet [Fellbaum, (1998, ed.), an English language electronic lexical database.

For the words in the datasets, we generate static word embeddings using the trained word embedding models (Word2Vec, GloVe and FastText). Each word embedding for a word is of dimension 300 and is used as feature input to train individual *Gender classifiers* (*w2v_classifier*, *ft_classifier* and *glove_classifier*) in predicting one of the three target labels (*male*, *female* or *neutral*). In the subsequent sections, we provide more details regarding the datasets, training classifier models and evaluation of classifier models.

Datasets The Multi-Dimensional Gender (MDGender) Bias Classification dataset is based on a general framework that decomposes gender bias in text along several pragmatic and semantic dimensions [Dinan et al., 2020]. It contains eight large scale datasets annotated for gender information, along with a list of gendered names (*names_gender*) and a list of gendered words (*gendered_words*) in English [Dinan et al., 2020]. Our work derives the list of gendered names and gendered words which provides us with a large list of names and words, along with their respective gender labels.

To account for words that are neither associated with *female* or *male*,

	Before		After	
Male	13,837	31.1%	8,921	33.3%
Female	21,707	48.8%	8,921	33.3%
Neutral	8,921	20.1%	8,921	33.3%
Total	44,465		26,763	
Training split			18,734	70%
Testing split			8,029	30%

Table 3.5: A summary of the dataset used in training the *Gender classifiers* before and after label balancing

we derive a list of neutral words from the WordNet (Fellbaum [(1998, ed.)] dataset. The authors of [Fellbaum, (1998, ed.)] tag English words with the natural gender of the person or type of person the word refers to and acts a source for gender-neutral words. Although this dataset also contains additional metadata, only the words and their respective gender labels are used in this work. An overview of the datasets is provided in table 3.4.

A combined list of 44,465 words were gathered from the two different source (MDGender and WordNet), along with their associated gender (male, female and neutral). Before using this combined list for training gender classifiers, the words in the list are subjected to simple preprocessing steps that involved converting all the names to lowercase, removing punctuation marks and digits, tokenizing the words and finally generating 3 different type of word embeddings (Word2Vec, GloVe and FastText) for each token.

Label Balancing As seen in table 3.4, the data instances of *neutral* label class and *male* label class in the training dataset is much less compared to the *female* class. Such an imbalance could be because fewer words in the MDDGender and WordNet datasets have been associated to male or neutral class. In any case, label imbalance like this could lead to the models performing poorly on the under-represented class compared to the majority classes. In order to balance the label classes, there are two commonly followed approaches - *oversampling* and *undersampling*. In oversampling, artificial data points are augmented to the under-represented class to balance the label distribution. In undersampling, data points are removed from the majority classes to balance them with the minority class.

While *Oversampling* could lead to overfitted classifier models due to the repetition of the minority class, *Undersampling* leads to loss of information as we drop some data points. We follow the undersampling approach and remove data points belonging to the majority classes *female* and *male*, since lesser data points would be computationally faster.

We represent the words with their word embeddings generated from 3 different word embedding models as feature inputs and target classes "male", "female" and "neutral" are mapped to 0, 1 and 2 respectively as class label representation. We then used the *train-test split* provided by *sklearn* [Pedregosa et al., 2011] to split the dataset into two parts, with 70% used as the training set and 30% for the testing set the gender classifier as shown in Table 3.5.

Baseline Classifiers We evaluated the performance of the baseline classifiers using the *sklearn DummyClassifier* class. This classifier serves as a simple baseline to compare against other more complex classifiers. The specific behaviour of the baseline is selected with the *strategy* parameter. As part of this work, we follow four strategies *most_frequent*, *stratified*, *uniform*, and *constant*, as described below.

- The *most_frequent* strategy predicts the most frequent class in the training set.
- The *stratified* strategy predicts the class based on the class distribution in the training set.
- The *uniform* strategy predicts the class uniformly at random.
- The *constant* strategy predicts a constant class label that can be specified.

We evaluate the *DummyClassifier* with all the above mentioned strategies, trained with 3 different word embeddings. With this as the baseline, we also consider two popular supervised classification algorithms Support Vector Machines (SVM, described in section 2.6) and Random Forest (RF) for the multi-class gender classification task at hand.

Support Vector Machines We follow the SVM implementation provided by the authors of Pedregosa et al. [2011]. SVM doesn't support multi-class classification natively. It supports binary classification and in the case of multi-class classification, the same principle is utilized after breaking down the multi-classification problem into multiple binary classification problems [Pedregosa et al., 2011].

SVM in a multi-class classification setting provides two approaches, *one-vs-one* (OVO) scheme where multi-class classification problem is broken down into multiple binary classification problems. A binary classifier is defined for each pair of classes. Whereas one-vs-rest (OVR) approach assigns one binary classifier per class and predicts based on membership to that class or

not [Hearst et al., 1998]. All the hyperparameters are set to default values provided by sklearn [Pedregosa et al., 2011] and a one-vs-rest (OVR) approach is followed to train a *base-svm* classifier using the training split of the dataset.

Random Forest A random forest fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting [Breiman, 2001]. We follow the RF classifier implementation provided by sklearn [Pedregosa et al., 2011] and train a *base-rf* classifier by choosing default values for all the hyperparameters.

After training *base-svm* and *base-rf* classifiers on the training split of the dataset, we identify a list of Hyperparameter combinations (for both SVM and RF classifiers) and evaluate these hyperparameter combinations using *GridSearchCV*. More details on the list of hyperparameters and classifier optimization are provided in the following section.

3.4.1 Hyperparameter optimization

Both SVM and RF classifiers have a number of hyperparameters that can significantly impact their performance on a given task. Unlike the internal parameters (weights, coefficients, etc.) that the algorithm automatically optimizes during model training, hyperparameters are model characteristics (e.g., Regularization parameter C for SVM and the number of estimators for an ensemble model like RF) that are set in advance. Choosing the right hyperparameters for an SVM classifier can be a challenging task, and an inappropriate selection of hyperparameters can result in a poorly performing model [Hearst et al., 1998]. For example, if the regularization parameter (C) is set too low, the model may overfit the training data, while if it is set too high, the model may underfit the data [Hearst et al., 1998]. Similarly, the choice of the number of estimators ($n_estimators$) and the maximum number of branches in each decision tree (max_depth) can also have a significant impact on the RF model’s performance.

We resort to one of the commonly used and effective hyperparameter tuning methods, *Grid Search* to find the best set of hyperparameters for both types of classifiers (SVM and RF). Grid search is a technique that involves creating a grid of all possible combinations of hyperparameter values and then training and evaluating a model for each combination. We use *GridSearchCV* provided by sklearn [Pedregosa et al., 2011], to hyperparameter tune both types of classifiers.

In the case of SVM classifiers, we identify the following hyperparameters as crucial for the model’s performance and define a range of values for each

of these hyperparameters and finally tune the SVM model using a grid search algorithm.

- C (Regularisation): C is the penalty parameter, which represents the number of misclassifications. This misclassification tells the SVM optimisation how much error is bearable.
- Gamma: Gamma defines how far influences the calculation of a plausible line of separation. When gamma is set higher data points closer will have a higher influence on the decision boundary and low gamma value means far away data points also be considered to get the decision boundary.
- Kernel: The kernel function is a crucial hyperparameter that determines how the input data is transformed before the SVM model is trained. The kernel parameter in SVM is used to specify the type of kernel function to use. The kernel function takes the input data and maps it to a higher-dimensional feature space, where the classes are more separable by a decision boundary.

```
Parameter grid used in Hyperparameter tuning
SVM classifier models
{
    C : [ 0.1, 1, 10, 100],
    gamma : [ 0.001, 0.01, 0.1, 1],
    kernel : [ 'rbf', 'poly']
}
```

For Random Forest classifiers we identify the following hyperparameters that could have an influence on the model's performance and define the value ranges used in our work.

- n_estimators: This hyperparameter specifies the number of decision trees to be used in the Random Forest [Pedregosa et al., 2011]. This is an important hyperparameter as the increase in the number of trees can increase the accuracy of the model, but at the same time, it can also increase the training time [Breiman, 2001].
- criterion: This hyperparameter is used to specify the quality of the split. The supported criteria are "gini" for the Gini impurity and "entropy" for the information gain [Pedregosa et al., 2011].

Hyperparameter Combinations for SVM classifiers				
Features	Hyperparameters			F_1 -score
	C (Regularisation)	Gamma	Kernel	
Word2Vec	0.1	0.1	Poly	0.6349
	1	0.1	Poly	0.6733
	10	0.1	RBF	0.6554
	100	0.01	Poly	0.7798
GloVe	0.1	0.01	Poly	0.6515
	10	0.01	RBF	0.6361
	100	0.01	Poly	0.6908
	100	0.001	RBF	0.6201
FastText	10	0.01	Poly	0.7024
	1	0.1	RBF	0.7512
	10	0.1	RBF	0.7188
	10	0.01	RBF	0.8279

Table 3.6: Four best hyperparameter combinations for different SVM based gender classifiers and the corresponding macro-averaged F_1 -score during cross-validation. Best performing classifiers are marked in bold.

- `max_depth`: This hyperparameter specifies the maximum depth of each decision tree. If this value is too large, the model may overfit. If it is too small, the model may underfit [Breiman, 2001].
- `max_features`: This decides the number of features to be considered when looking for the best split [Pedregosa et al., 2011].

```

Parameter grid used in Hyperparameter tuning
RF classifier models
{
    n_estimators : [100, 200, 300, 400],
    criterion : ['gini', 'entropy'],
    max_depth : [5, 10, 15, 20],
    max_features : ['sqrt', 'log2']
}

```

In the case of both SVM and RF classifiers, the scikit-learn documentation [Pedregosa et al., 2011] provides a full list of available hyperparameters. For the rest of these hyperparameters, we will use the default values defined by scikit-learn. With a default cross-validation strategy parameter ‘cv’ (default value = 5) in *GridSearchCV*, the training data is divided into five folds. The

Hyperparameter Combinations for RF classifiers					
Features	Hyperparameters				F_1 -score
	estimators	criterion	max depth	max features	
Word2Vec	200	gini	10	sqrt	0.7026
	200	entropy	15	sqrt	0.7091
	400	gini	20	log2	0.7136
	400	entropy	20	log2	0.7190
GloVe	100	entropy	5	log2	0.5961
	200	gini	10	log2	0.6306
	400	gini	10	sqrt	0.6788
	400	entropy	15	sqrt	0.7641
FastText	100	gini	5	log2	0.6502
	200	gini	10	log2	0.7102
	200	entropy	10	log2	0.7040
	200	entropy	10	log2	0.7093

Table 3.7: Four best hyperparameter combinations for different RF-based gender classifiers and the corresponding macro-averaged F_1 -score during cross-validation. Entries for best performing classifiers are marked in bold.

classifier is trained five times, where four different folds of data are used for training each time and the remaining one fold for validation. The validation scores of the five runs are averaged, and the hyperparameter combination with the best score is chosen as the best parameter. The evaluation metric used for evaluating the combinations is *macro-F1* score since this would consider the performance of all target classes equally. After performing a total of 160 SVM model fits and 400 RF model fits, we compare the performance of classifiers to find the best performing gender classifiers on the training data (trained on 3 different word embedding feature inputs). Table 3.6 shows the best performing hyperparameter combinations for SVM based classifiers, trained on 3 different word embeddings (Word2Vec, GloVe and FastText), while Table 3.7 shows the best-performing combinations for the RF based classifiers identified during grid search. The best parameters are highlighted in both tables. We performed the above described classifier training process, using the training split of the dataset with feature input generated from 3 different word embeddings. From the cross-validation results displayed in Table 3.6, we found that SVM based classifier performs the best for Word2Vec (F_1 score-0.7798) and Fasttext word embeddings (F_1 score-0.8279). In the case of GloVe based embeddings, the Random Forest classifier (F_1 score-0.7641) is found to be the best performing gender classifier.

For each word embedding type as feature input, we compared all the trained

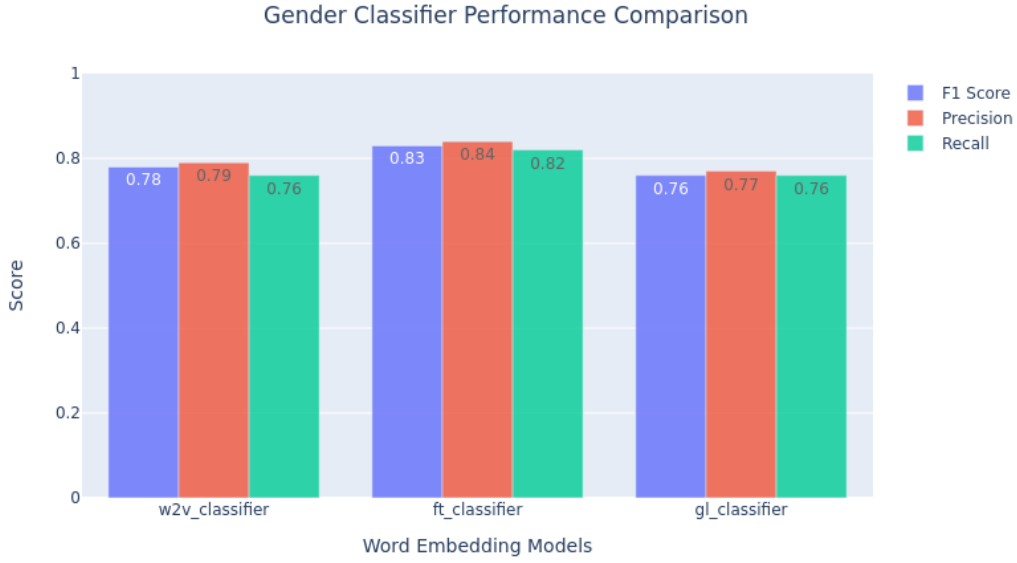
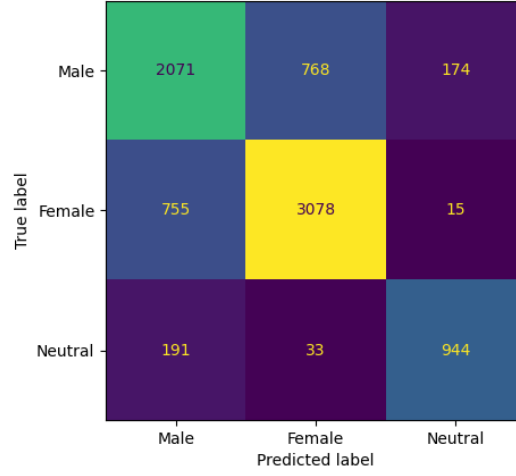


Figure 3.3: Overview of chosen *Gender classifier* performances in terms of F_1 score, Precision and Recall. The *w2v_classifier*, *ft_classifier* and *gl_classifier* are trained on Word2Vec, FastText embeddings and GloVe model embeddings respectively.

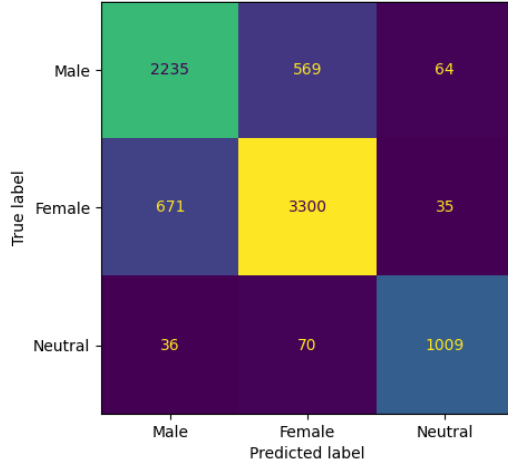
classifiers and chose the best performing classifier as our *Gender classifiers* (*w2v_classifier* - SVM, *ft_classifier* - SVM and *glove_classifier* - RF) and an overview of individual F_1 score, precision and recall is shown in Figure 3.3. These classifiers are then evaluated on the held-out Testing split of the dataset (Table 3.5) and are further used in our proposed methods in inferring gender labels for words in the *Pile* dataset. The evaluation results of each of these classifiers are described in the following section.

Gender Classifier Evaluation We evaluated three gender classifiers (each trained on a type of word embedding as feature input), using the *Testing split* of the dataset. The testing split of the dataset consists of a mix of *male*, *female* and *neutral* words with a total of 8,029 words.

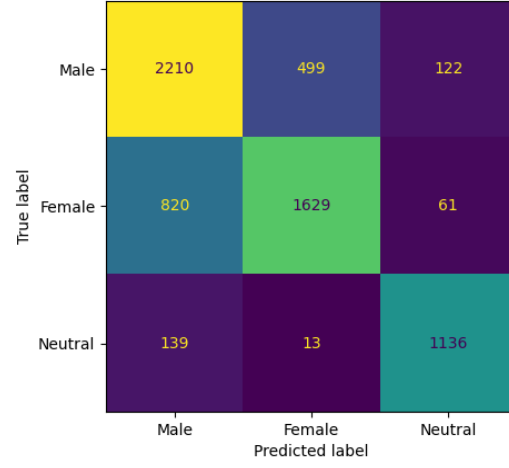
While The F1-Score, Precision and Recall are relevant metrics to evaluate the overall performance of the classifiers, we also make use of the confusion matrix (described in Section 2.2.1.1) to illustrate the number of correctly and incorrectly classified data instances. Figure 3.4 shows the confusion matrices for the three chosen gender classifiers and provides the number of true positives, true negatives, false positives and false negatives on the testing split instances. With this information, we compute the False Positive Rate (FPR)



(a) Word2Vec Classifier



(b) FastText Classifier



(c) GloVe Classifier

Figure 3.4: Confusion Matrices for best performing Gender Classifiers

Gender Classifier	Male		Female		Neutral	
	FPR	FNR	FPR	FNR	FPR	FNR
w2v_classifier	0.234	0.219	0.454	0.206	0.181	0.207
ft_classifier	0.343	0.256	0.395	0.211	0.096	0.067
glove_classifier	0.302	0.299	0.407	0.251	0.168	0.017

Table 3.8: Caption

and False Negative Rate (FNR) for each class, to assess more about the individual classifier’s performance. False Negative Rate (FNR) is the ratio of positive class (e.g., male or female or neutral) instances predicted as a negative class (two classes other than the class chosen as a positive class). Similarly, the False Positive Rate is the ratio of negative class instances predicted as a positive class. Table 3.8 shows the FPR and FNR for each class (male, female and neutral) for the three gender classifiers.

In general, the FPR rates for *male* class are deemed low, although FNR for all 3 classifiers ranging between 0.25 to 0.35 indicates that 25% to 35% of the instances that are actually *female* or *neutral* are incorrectly classified as *male*. For all three classifiers, we also observed that the FPR for *female* class is higher relative to the other classes. In particular to word2vec classifier, almost 45% of the instances that are actually *male* or *neutral* are incorrectly classified as female. This could be problematic when it is finally used in predicting class labels for words. This is seen as a drawback of the trained classifier and needs further investigation into model parameters in order to improve its performance. However, in case of *neutral* class prediction, all three classifiers perform well in terms of FPR and FNR as shown in Table 3.8. Overall, for all three classifiers, more instances of *male* and *neutral* words are accurately classified over *female* words.

These developed gender classifiers are further used in predicting gender labels for words in the *Pile* dataset as part of our approaches for word-list creation. We look at each of the word-list creation approaches in the following sections.

3.5 Frequency First

We have seen that the process of word-list selection for measuring bias (in particular gender bias) is not well understood and more often researchers leverage the word-lists from past works in psychology and social sciences, without taking into account the underlying dataset (section 3.3.1). From the identified factors that could have an influence on word-lists, as described in section 3.3.1, we focus on the lexical factors of individual terms like frequency (number of word occurrences in the underlying data) as part of our proposed *Frequency First* method.

Social science literature suggests that past experiences and associations can have a powerful and often unconscious influence on present-day judgments and behaviour of humans, which may be reflected in the frequency of certain words and their associations with genders in language data [Jacoby et al., 2004]. For instance, if a human participant has seen a name more frequently (i.e., at least

on two separate occasions), they judge that name to be more famous than a name they have seen less frequently, leading to biased judgements against more frequent terms. Applying the same principles to word embeddings, the authors of Caliskan et al. [2022] argue that the relative frequency of words in the training data can lead to certain associations between words and genders in the resulting embeddings, which can perpetuate gender stereotypes and biases. That is if a given social group (e.g., men) has a more frequent representation than another group (e.g., women), the more frequent group will come to shape what we perceive as default.

We adapt these insights in our method for creating word-lists to represent social groups (*male* and *female*), where we intend to identify words associated with each social group and choose words that occur more frequently in the dataset to represent each social group. As part of *Frequency First* method, we perform the following steps to create word-lists to represent the social groups *male* and *female*, which are further used in examining two exemplary gender stereotype studies (described in section 3.3).

Step 1: Inferring Gender labels First goal is to identify all the words associated with ‘male’ and ‘female’ groups, by classifying all the unique words into 3 gender groups (*male*, *female* and *neutral*) using the *Gender classifier* trained previously. The preprocessed *Pile* dataset which was used to train 3 Word Embedding models, consists of a total of 8,849,888 unique English language tokens. We leverage the chosen *Gender classifier* for each word embedding type, to infer one of the gender class labels for all the words in the word embedding model vocabulary.

Step 2: Ordering by Frequency Having classified words in to one of the gender groups (*male*, *female* and *neutral*), we find all words associated with male and female groups. We then get the count of word occurrences (word frequency) in the *Pile* dataset, for all words in male and female gender groups. We then sort words in each of these groups in the descending order of the word frequency.

Step 3: Creating Target Sets After creating word-lists with male and female associated words, sorted in descending order in terms of word frequency, we create target sets by choosing only the top frequent words associated with each target group. We create 3 subsets of target sets by choosing top 1k most frequent words (*top1k_male* and *top1k_female*), top 5k most frequent words (*top5k_male* and *top5k_female*) and top 10k most frequent words (*top10k_male* and *top10k_female*) associated with each target group.

With the above 3 steps, the *Frequency First* provides a data-driven method to create target sets and these created target sets are evaluated using Social bias metrics (WEAT and RNSB) to examine two stereotypes (discussed in 3.3) in trained word embeddings. The WEAT and RNSB scores are then compared to the effect size scores of the benchmark IAT studies conducted to examine the two stereotypes.

3.6 POS Filter

The process of classifying words into their parts of speech and labelling them accordingly is known as part-of-speech tagging or POS-tagging. Parts of speech are also known as *word classes* or *lexical categories*. A set of all POS tags used in a corpus is called a POS tagset. A high-level analysis of the POS tagsets shows that the majority of tagsets are very fine-grained and distinguishes 48 different POS tags (e.g., *VB* - Verb, *VBD* - Verb in its past tense, *NN* - Singular Nouns, *NNS* - Plural Nouns etc) [Marcus et al., 2002]. Here we adopt a POS tagset provided by Smith and Eisner [2005], who defined a collapsed set of 17 English POS tags (instead of 48 tags) that has subsequently been adopted by many POS induction works. These 17 POS tags are known as Universal POS tags and are broadly classified into *Open* class, *Closed* class and *Other* class of words [Smith and Eisner, 2005], as shown in table 3.9. Open classes typically comprise words that carry meaning, they have the ability to create new words and expand their membership through the introduction of new words [Smith and Eisner, 2005]. Nouns, verbs, adjectives, and adverbs are examples of open classes. Closed classes, on the other hand, are comprised of function words that have a relatively fixed membership and do not typically create new words [Smith and Eisner, 2005]. Determiners, prepositions, pronouns, conjunctions, and auxiliary verbs are examples of closed classes. Finally, other classes are those that do not fit neatly into the open or closed categories. This includes interjections, symbols, and other grammatical particles.

The authors of Caliskan et al. [2022] examined the parts-of-speech (POS) distribution of words associated with *male* and *female* groups and suggests that words associated with "men" are more action words or Verbs (e.g., *fight*, *overpower*) and words associated with "women" are typically Adjectives or Adverbs (e.g., *beauty*, *giving*, *emotional*). Thus in order to account for such gender biases in word embeddings that permeate through parts-of-speech, we aim to represent "male" and "female" groups by words that not only belong to a few stereotypical parts-of-speech types (such as *verbs* or *adjectives*) but rather include associated words equally from all the relevant parts-of-speech types that help to better represent the social groups.

Universal POS Tags		
Open Class	Closed Class	Other
ADJ, Adjective	ADP, Adposition	PUNCT, Punctuation
ADV, Adverb	AUX, Auxiliary	SYM, Symbols
INTJ, Interjection	CCONJ, Coordinating Conjunction	X, Other
NOUN, Noun	DET, Determiners	
PROPN, Proper Noun	NUM, Numerals	
VERB, Verbs	PART, Particle	
	PRON, Pronoun	
	SCONJ, Subordinating Conjunction	

Table 3.9: List of all the Universal POS tags defined by Smith and Eisner [2005], broadly classified into Open, Closed and Other class of words. POS tags relevant to our work are marked in bold.

With this context, we propose the *POS filter* method to create word-lists (target sets for representing gender groups) by identifying all the relevant word classes that could improve the representation of each gender group. In each identified relevant class of words, we classify the words using Gender classifier into "male", "female" and "neutral" classes. We then choose the most frequent words for two gender classes (*male and female*) of words as entries into respective target sets. More details of the proposed method are described in the following steps.

Step 1: POS Tagging: Following the work of Caliskan et al. [2022], we leverage a state-of-the-art English language parts-of-speech tagger *upos-english* provided by *flair* [Akbik et al., 2019]. *upos-english* tagger predicts a POS tag for a word from 17 Universal parts-of-speech categories. All the words in the *Pile* dataset are tagged into 17 word classes (POS tags).

Now we choose words from specific word classes that help in representing a social group (male or female). We manually inspected all the word classes to identify that not all classes of words help in representing a social group. For e.g., words in the classes like ADP (e.g., *in, to, during*), AUX (e.g., *has, is, will*), INTJ (e.g., *psst, ouch, bravo*), CCONJ (e.g., *and, or, but*), NUM (e.g., 5, 2014, 3.141) and others were typically found to be comprising of function words. Such function words are not gendered and thus do not contribute to representing a particular gender based social group. However words from classes such as NOUN (e.g., *boy, girl, tree*), PROPN (e.g., *Mary, John, London*), VERB (e.g., *run, hit, cook*), PRON (e.g., *he, she, it*) and ADJ (e.g., *old, young, smart*) comprised of gendered words that could represent a gender group and thus are found to be relevant to our work. We created individual

word-lists by grouping words in 5 POS classes (ADJ, NOUN, PROPN, PRON and VERB) by ignoring all the other words.

Step 2: Inferring Gender labels and Frequency Ordering: Once we have word-lists associated with POS tags under consideration, we then find male and female associated words for each POS category. For this purpose, we leverage the Gender classifiers (trained previously) to infer gender labels (*male, female or neutral*) for word-list in each relevant POS category. From each POS tag and gender group, we choose words that occur frequently in the dataset by sorting each word list in descending order of occurrence count (word frequency). At the end of this step, we create sorted word-lists of male and female associated words in each of the 5 POS tags relevant to this work, ranked by descending order of word frequency.

Step 3: Creating Target Sets: In this step, we create 2 target sets with top frequent male and female words by choosing an equal number of words in each of the relevant POS tag word-lists created in Step 2. We created 3 subsets of target sets for each gender group, male - *pos_top1k_male*, *pos_top5k_male*, *pos_top10k_male* and female - *pos_top1k_female*, *pos_top5k_female*, *pos_top10k_female*, which allows us to evaluate target sets with relevant word classes and also varying word frequencies.

These created target sets are further used to evaluate the gender bias in trained word embedding models to examine the two stereotypes (described in section 3.2).

3.7 Semantic Word Clustering

Along with lexical factors like word frequency and POS of a word, Definitional factors like reductive and imprecise definitions could influence the word-lists representing target social group [Antoniak and Mimno, 2021]. Word-lists have the potential to oversimplify and categorize life experiences in a way that reduces complex experiences to basic categories (*Reductive Definitions*) [Antoniak and Mimno, 2021]. When names are used as placeholders for complex concepts like race or gender, or when we simplify gender to just two categories (male and female), it can lead to reduced understanding of the underlying data [Nguyen et al., 2014b]. If the target social groups are not clearly defined, the resulting word-lists may become too general and encompass multiple concepts (Imprecise Definitions) [Antoniak and Mimno, 2021]. These word-lists when used to measure social bias (e.g., gender bias) could lead to incorrect bias measurements.

Table 3.10: List of parameters and their set values used for training k-means clustering model provided by Pedregosa et al. [2011].

heightParameters	Value
init (Method of initialization)	Target bias concepts as initial centroids
n_init (Number of runs with different centroid seeds)	10
max_iter (Maximum number of iterations)	300
algorithm (K-means algorithm to use)	<i>lloyd</i>

To address these factors, we present a clustering approach to group words that are similar in meaning into clusters (target bias concepts as cluster centroids), classify words in each cluster into gender groups (*male* and *female*), sort words in each cluster in the descending order of their semantic similarity (based on cosine similarity of word embeddings) to cluster centroid and finally choose top words from each gender group in each cluster as entries into target sets. For e.g., In the case of *Career-Gender* study (described in section 3.3), we group all the words into clusters by choosing a combined list of *career* (*executive, management, professional etc*) and *family* (*home, parents, children, family etc*) related terms as the cluster centroids, classify words in each cluster into *male* and *female* groups and use these words as target sets in gender bias measurement. More details on individual processing steps are described in the following.

Step 1: Semantic Clustering In this step we create word-lists with *male* and *female* associated words for each target bias concept term. Words in the word embedding models are clustered using an implementation of *k-Means* algorithm (described in section 2.2.1.2) provided by [Pedregosa et al., 2011]. Since word embedding models represent words as embeddings (vectors) whose positions relative to each other represent the words' semantic and physical relationships in the dataset, *k-Means* clustering is a relatively effective method of topically clustering corpora. A combined list of all the target bias concept terms (Career and Family terms for Career-Gender IAT; Maths and Arts related terms for Maths-vs-Arts IAT) are chosen as initial cluster centroids. All the other parameters are initialized with default values provided by Pedregosa et al. [2011] as shown in table 3.10. This step results in word-lists, one for each cluster formed. These word-lists are used in further steps to finally create target sets for social groups (gender based groups, *male* and *female*) that are

semantically closer to target bias concepts.

Step 2: Inferring Gender Labels and Similarity Ordering We predict gender labels (*male*, *female* and *neutral*) for all the words in the word-lists created in Step 1, using previously trained *Gender classifiers*. We then compute cosine similarity (cosine similarity is a measure of similarity between two non-zero vectors defined in an inner product space.) between each word and their cluster centroid. Words in each male and female associated word-list are then sorted in the descending order of cosine similarities with their centroids. At the end of this step, we get gendered word-lists for each cluster ranked by a similarity measure.

Step 3: Creating Target Sets The gendered word-lists created in Step 2 are now used in creating target sets for representing male and female social groups. We choose only top ranked male and female associated words equally from each cluster to create target sets. These sets are used in bias experiments to examine the stereotypes described in two IAT studies (described in section 3.3).

Semantic Word Clustering method for creating target sets that are representative of gender groups (*male* and *female*) is evaluated by using created target sets for measuring gender bias in three word embeddings models. Experiments conducted and their evaluation details are provided in chapter 4.

Chapter 4

Experiments and Evaluation

In this chapter, we evaluate *Frequency First*, *POS Filter* and *Semantic Word Clustering* approaches for creating word-lists (target sets for representing *male* and *female* groups) as described in Chapter 3. Using the newly created word-lists from each developed method, we evaluate 3 different word embedding models (*Word2Vec*, *GloVe* and *FastText*) trained on the *Pile* dataset, for *Gender bias* using word embedding based bias metrics (*WEAT* and *RNSB*). We report the WEAT (*effect_size* and *p-value*) and RNSB scores for each experiment conducted. We used the WEAT and RNSB implementation of the open-source Word Embedding Fairness Evaluation framework (WEFE) by Badilla et al. [2020].

For our experiments, we consider two benchmark IAT studies by Greenwald et al. [1998], designed to examine gender stereotypes such as, 1) *Career vs Family* IAT (stereotype: *men* are more associated with professional job roles while *women* are more associated with traditional family roles) and 2) *Math vs Arts* Gender IAT (stereotype: *female* terms are more associated with arts than mathematics related academic domains, compared to *male* terms).

Baseline: Replicating IAT Studies Two IAT studies under consideration (Career vs Family and Math vs Arts IAT), which were conducted on human participants (38,797 and 28,108 participants respectively) reported IAT effect sizes of 0.72 ($p\text{-value} < 10^{-2}$) and 0.82 ($p\text{-value} < 10^{-2}$) respectively [Greenwald et al., 1998]. These two studies were also replicated by the authors of the original WEAT paper [Caliskan et al., 2016], using WEAT (*effect size*) on GloVe word embeddings trained on a corpus of ordinary language found on the internet. The authors [Caliskan et al., 2016] used the same stimuli (sets of target and attribute words) as in IAT (described in Section 3.3), to report that word embeddings also exhibit implicit associations (For e.g., *female* terms with *family* related terms, *male* terms with *career* terms) similar to that of human subjects in the two IAT studies. They reported WEAT effect sizes of

IAT Studies	Embeddings	WEAT		RNSB
		<i>effect size</i>	<i>p-value</i>	
Career vs Family	Word2Vec	1.5235	0.025	0.0564
	FastText	1.7279	$< 10^{-3}$	0.2369
	GloVe	1.7493	0.001	0.2654
Math vs Arts	Word2Vec	0.7255	0.057	0.0912
	FastText	0.5082	0.050	0.0574
	GloVe	1.1857	$< 10^{-3}$	0.1843

Table 4.1: Results of Baseline evaluation, validating the results of the original WEAT paper [Caliskan et al., 2016] for examining two gender stereotypes in Word2Vec, GloVe and FastText models trained on the *Pile* dataset. The *target* and *attribute* sets provided by Nosek et al. [2002a] were used here. We report *p-values*, *effect size* values for WEAT and RNSB metric scores.

1.82 for *Career vs Family* study and 1.06 for *Math vs Arts* study with *p-value* $< 10^{-2}$ in both cases.

As part of the evaluation, we initially replicate the two benchmark IAT experiments using the same stimuli provided by the authors of [Greenwald et al., 1998]. We use both WEAT and RNSB to evaluate the implicit biases in our word embedding models (Word2Vec, GloVe and FastText). The effect sizes resulting from these experiments are regarded as a *Baseline* reference and serve as a basis for comparison against the results obtained through our techniques.

Study 1 To examine the traditional job role stereotypes associated with gender groups, we aim to observe the association between target sets representing male and female groups and attribute sets representing traditional careers and domestic roles. As a *Baseline* reference for this study, we used target sets and attribute sets provided by the authors [Nosek et al., 2002a] and evaluated each type of word embedding model. Inspecting the target sets (*male* and *female*, given in Section 3.3), it can be seen that each gender group is represented using a short-list (8 terms) of names of people.

For the Word2Vec model, we reported a WEAT effect size of 1.5235 with a *p-value* of 0.025, showing a strong association between male terms and career terms and between female terms and family terms. Similarly, from the reported WEAT effect sizes for FastText (1.7279) and GloVe (1.7493) models, we observed strong associations for male terms with career terms and female terms with family related terms.

Approach	Target sets		Attribute Sets	
Frequency First	top1k_male	top1k_female	career	family
	top5k_male	top5k_female	career	family
	top10k_male	top10k_female	career	family
	top1k_male	top1k_female	math	arts
	top5k_male	top5k_female	math	arts
	top10k_male	top10k_female	math	arts
POS Filter	pos_top1k_male	pos_top1k_female	career	family
	pos_top5k_male	pos_top5k_female	career	family
	pos_top10k_male	pos_top10k_female	career	family
	pos_top1k_male	pos_top1k_female	math	arts
	pos_top5k_male	pos_top5k_female	math	arts
	pos_top10k_male	pos_top10k_female	math	arts
Semantic Word Clustering	cluster_male	cluster_female	career	family
	cluster_male	cluster_female	math	arts

Table 4.2: List of all the Gender bias experiments conducted for each Word embedding model trained on the Pile dataset, with the newly created target sets using approaches developed in this thesis work.

Study 2 In the second study we examine the gender stereotypes associated with *Arts* and *Mathematics* related academic domains. Here we observe the association between target sets representing male and female groups and attribute sets representing arts and math related terms. Target sets for this study comprise male (e.g., *he*, *him*, *boy*, *son* etc) and female (e.g., *she*, *her*, *sister*, *girl* etc) terms. Target and attribute sets provided by the authors [Nosek et al., 2002a] (refer Section 3.3) were then used to evaluate each word embedding model and the results are then referred to as *Baseline* for this study. The WEAT (effect size and p-values) and RNSB metric scores for the three word embedding models are shown in the Table 4.1.

In general, results from both studies indicate that all three word embedding models show implicit associations similar to that found in experiments conducted by Caliskan et al. [2016] (refer Table 4.1). As part of further experiments, we evaluate the newly created *target sets* for representing *male* and *female* groups) and *Attribute Sets* provided for the two IAT studies (described in Section 3.3). An overview of all the experiments performed and word-lists used for representing each of the two targets and attributes are listed in Table 4.2.

Word2Vec	<i>male</i>	<i>to, it, was, he, his, players, son, husband, demonstrate, god, performance, money, murphy, stanley ..</i>
	<i>female</i>	<i>the, of, and, summer, omega_, mrs, pink, her, females, mother, girls, flower, life, she, hair, karen, pregnant ..</i>
FastText	<i>male</i>	<i>to, was, he, his, son, husband, dj, examination, danger, power, dealer, golf, john, bristol, sibling ..</i>
	<i>female</i>	<i>the, of, and, ms, amy, julia, dancing, clothes, yu_, she, dress, moon, she, mother, dear, care, consistency ..</i>
GloVe	<i>male</i>	<i>to, it, was, he, his, cyrus, mike, dare, smart, husband, father, god, money, cycle, engineering..</i>
	<i>female</i>	<i>the, of, and, amy, karen, house, children, winter, her, awareness, disaster, secret, kiss, dances, care, gamma_ ..</i>

Table 4.3: A sample list of gender-associated words that were included in the tests for *Frequency First* approach. All the words in the *Pile* dataset were classified using three different *Gender classifiers*, each trained on a type of word embedding as feature input.

4.1 Frequency First

Frequency First approach is developed with the aim of representing target social groups (*male* and *female* in our case) with terms with higher occurrence since words occurring frequently will have the strongest influence on group representation [Caliskan et al., 2022]. Our approach addresses the challenge of creating target sets (to better represent gender-based social groups) by selecting the most frequently occurring words in the dataset, associated with each gender (*male* and *female*). Following the steps described in Section 3.5, we inferred gender labels for all the words in the *Pile* dataset using a Gender Classifier, to classify them into *male*, *female* and *neutral* word groups. We sorted each of these word groups in the descending order of word occurrences in the dataset (word frequency). Finally, frequently occurring words were chosen from the two gender groups (*male*, *female*) as entries into target sets, which are further used in the two above-mentioned gender bias experiments.

In order to also investigate the influence of varying frequency of words, we created three sets of target sets for *male* and *female* groups, by choosing 1,000 (*top1k_male*, *top1k_female*), 5,000 (*top5k_male*, *top5k_female*) and 10,000 (*top10k_male*, *top10k_female*) most frequently occurring words from each group as entries into the target sets. A sample of the gender-associated words from the newly created word-lists for each word embedding type is provided in Table 4.3 (refer to A.1 for a full list of *top1k_male* and *top1k_female* target sets used in Word2Vec evaluation). These target sets were then used in

Career VS Family					
Embeddings	Target sets		WEAT		RNSB
			<i>effect size</i>	<i>p-value</i>	
Word2Vec	top1k_male	top1k_female	0.3002	$< 10^{-3}$	0.1028
	top5k_male	top5k_female	0.4227	$< 10^{-3}$	0.0940
	top10k_male	top10k_female	0.4138	0.0376	0.0901
FastText	top1k_male	top1k_female	0.3382	$< 10^{-3}$	0.1570
	top5k_male	top5k_female	0.4929	0.0451	0.1730
	top10k_male	top10k_female	0.4819	0.0576	0.1631
GloVe	top1k_male	top1k_female	0.3732	$< 10^{-3}$	0.2009
	top5k_male	top5k_female	0.5129	0.076	0.2381
	top10k_male	top10k_female	0.5231	1.123	0.2401

Table 4.4: WEAT and RNSB results of Word2Vec, FastText and GloVe models trained on the *Pile* dataset, in examining *Career and Family* related stereotypes associated with *male* and *female* groups. Target sets (male, female) created using *Frequency First* approach and attribute sets (career, family) provided by [Nosek et al., 2002a] were used in all the experiments. WEAT effect sizes and RNSB scores were reported, where only WEAT effect sizes with a *p-value* < 0.05 are considered statistically significant values.

evaluating three word embedding models for two different gender stereotypes by computing WEAT (*effect size*, *p-value*) and RNSB scores.

Study 1 We performed three tests using the newly created target sets for *male* and *female* groups and the IAT attribute sets provided by the authors of [Nosek et al., 2002a], to investigate the career related stereotypes for three types of trained word embeddings. WEAT (*effect size* and *p-value*) and RNSB scores for all the tests for Study 1 are presented in the Table 4.4.

In the case of *top1k_male* and *top1k_female* target sets, word2vec embeddings resulted in WEAT size of 0.3002 (with a *p-value* $< 10^{-3}$) indicating an association between male sets with career terms and female sets with family terms. The word2vec embeddings also show a similar trend in the case where we used 5,000 and 10,000 most frequently occurring *male* and *female* associated terms as target sets, with reported WEAT effect size of 0.4227 and 0.4138 respectively. However, it can be noticed that there is a small increase in the measured WEAT effect sizes with target sets comprising 5,000 (+0.1225) and 10,000 (+0.1136) terms from the first case (target sets with 1,000 terms), indicating a slightly stronger association between the target sets and the attribute sets. The reported RNSB scores for the word2vec model also show an increase in the reported scores (between +0.05 and +0.15) for all three cases

Math VS Arts					
Embeddings	Target sets		WEAT		RNSB
			<i>effect size</i>	<i>p-value</i>	
Word2Vec	top1k_male	top1k_female	0.1691	$< 10^{-3}$	0.2229
	top5k_male	top5k_female	0.3327	$< 10^{-3}$	0.1462
	top10k_male	top10k_female	0.2958	0.105	0.1212
FastText	top1k_male	top1k_female	0.1522	$< 10^{-3}$	0.2520
	top5k_male	top5k_female	0.3055	$< 10^{-2}$	0.1898
	top10k_male	top10k_female	0.2650	0.085	0.1884
GloVe	top1k_male	top1k_female	0.1763	$< 10^{-2}$	0.2601
	top5k_male	top5k_female	0.2890	0.13	0.2128
	top10k_male	top10k_female	0.2705	1.005	0.2023

Table 4.5: Gender bias experiments to examine stereotypes related to Math and Arts domains in Word2Vec, FastText, and GloVe models trained on the Pile dataset. Target sets, created using the *Frequency First* approach, and attribute sets (math, arts) from the IAT studies were used. Reported WEAT effect sizes and RNSB scores, where only WEAT effect sizes ($p\text{-value} < 0.05$) were considered statistically significant.

from the Baseline tests (refer Table 4.1). The FastText model results in significantly lower WEAT *effect size* values in all three cases when compared to the Baseline test results. The RNSB scores reported for the FastText model also indicate a negative sentiment towards the target set terms. The GloVe model results in a WEAT effect size of 0.3732 (with a $p\text{-value} < 10^{-3}$) for the case with 1,000 terms in the target sets, showing a decrease in WEAT *effect size* values from the Baseline test results. However, in the case of 5000 and 10000 terms in the target sets, GloVe embeddings resulted in WEAT *effect sizes* with $p\text{-values}$ greater than 0.05 which are considered statistically not significant in our work.

Study 2 Here we performed tests to examine the gender stereotypes associated with *math* and *arts* related domains, using the newly created word-lists. All the results for this study are presented in Table 4.5. In the case of target sets with 1,000 most frequently occurring terms, all three word embedding models reported WEAT *effect sizes* showing a decrease in the measured bias as compared to respective Baseline results (refer Table 4.1). For target sets with 5,000 terms, Word2Vec showed an increase of 0.4 in WEAT effect size value as compared to the case with 1000 terms in target sets. However, the other two models resulted in WEAT *effect sizes* with $p\text{-values}$ greater than 0.05 and hence were considered to be statistically not significant. Similarly, in

Word2Vec	male	NOUN	husband, brother friends, boys, science ..
		PROPN	paul, adam, john, david, william, henry ..
		PRON	he, his, me, them, him, us, myself ..
		ADJ	smart, willing, sharp, timely, strategic ..
		VERB	getting, saying, investigated, parking ..
	female	NOUN	lady, girls, woman, parents, house ..
		PROPN	georgia, maria, yoshimi, asia, canada ..
		PRON	they, my, their, her, she, our, herself ..
		ADJ	beauty, graceful, smoothness ..
		VERB	get, dancing, asking, dreaming..

Table 4.6: A sample list of gender-associated words for each identified POS class that were included in the target sets created using the *POS Filter* approach. The words were then classified using *Gender classifier*, trained on Word2Vec embeddings.

the case of 10,000 terms for each target group, we observed that all the models produced results with *p-values* greater than 0.05. In general, the RNSB scores for all three models in all three cases were found to be similar to the Baseline results, indicating similar negative sentiment towards terms in the target sets.

Overall, the word-lists (target sets for *male* and *female* groups) that consisted of the most frequently occurring gender associated were used to evaluate two gender stereotypes in three custom trained word embedding models on the *Pile* dataset. The bias measurement results were significantly lower than that of the Baseline results but indicated a clear association between target sets and attribute sets in both the IAT studies. These target sets and the bias measurement results are further discussed in Section 4.4.

4.2 POS Filter

As described in Section 3.6, *POS Filter* approach aims to create word-lists to represent two gender groups (*male* and *female*) with words from POS classes (word classes) that are generally used to describe particular gender groups. Based on research by the authors of [Caliskan et al., 2022], we identified 5 classes of words, NOUN (Nouns, e.g., *mother, father, girl, boy etc*), PROPN (Proper Nouns, e.g., *amy, donna, john, chris etc*), PRON (Pronouns, e.g., *she, he, her, him etc*), ADJ (Adjectives, e.g., *beautiful, weak, smart, strong etc*) and VERB (Verbs, e.g., *cook, bake, build, run*) that are more often associated with a particular gender group. We find the most frequently occurring gendered and gender-associated words in each of the identified word classes. These words are then used as entries into the target sets to represent each gender group in

Career VS Family					
Embeddings	Target sets		WEAT		RNSB
			<i>effect size</i>	<i>p-value</i>	
Word2Vec	pos1k_male	pos1k_female	0.2864	$< 10^{-3}$	0.1018
	pos5k_male	pos5k_female	0.3537	$< 10^{-3}$	0.0867
	pos10k_male	pos10k_female	0.5537	$< 10^{-2}$	0.0179
FastText	pos1k_male	pos1k_female	0.3266	$< 10^{-3}$	0.1559
	pos5k_male	pos5k_female	0.4546	0.0312	0.1667
	pos10k_male	pos10k_female	0.6156	0.0576	0.2012
GloVe	pos1k_male	pos1k_female	0.3902	$< 10^{-2}$	0.0915
	pos5k_male	pos5k_female	0.4471	0.205	0.1982
	pos10k_male	pos10k_female	0.4129	0.561	0.2531

Table 4.7: Gender bias experiments to examine stereotypes related to Career and Family related roles in Word2Vec, FastText, and GloVe models trained on the Pile dataset. Target sets, created using the *POS Filter* approach, and attribute sets (*career*, *family*) from the IAT studies were used. Reported WEAT effect sizes and RNSB scores, where only WEAT effect sizes ($p\text{-value} < 0.05$) were considered statistically significant.

bias experiments.

Similar to the *Frequency First* approach, we created 3 sets of word-lists comprising 1,000 (*pos5k_male/female*), 5,000 (*pos5k_male/female*) and 10,000 (*pos10k_male/female*) most frequently occurring gender-associated words by choosing words from each identified word classes. A sample list of words from the *pos1k_male* and *pos1k_female* target sets used in evaluation of word2vec embeddings is provided in Table 4.6 (refer to A.2 for a full list of *top1k_male* and *top1k_female* target sets used in Word2Vec evaluation). These target sets are then used to examine two gender stereotypes (refer Section 3.3) in three word embedding models trained on the *Pile* dataset.

Study 1 Using the newly created word-lists, we examine the exemplary gender stereotype, that men (*male*) terms are more associated with *career* terms and women (*female* terms) are associated with traditional domestic roles (*family* terms). Results for all the performed tests are presented in Table 4.7. We observed that word2vec embeddings produced statistically significant WEAT *effect sizes* in all three cases, indicating a bias towards *male* group for the *career* and *family* categories. The RNSB scores for word2vec embeddings showed very low negative sentiment towards the target set terms in all three cases. Although there is a significant difference in the measured WEAT *effect sizes* for FastText embeddings with respect to its Baseline test results, the WEAT effect

Math VS Arts					
Embeddings	Target sets		WEAT		RNSB
			<i>effect size</i>	<i>p-value</i>	
Word2Vec	pos1k_male	pos1k_female	0.1593	$< 10^{-3}$	0.1772
	pos5k_male	pos5k_female	0.2996	$< 10^{-2}$	0.1267
	pos10k_male	pos10k_female	0.4410	0.0184	0.1390
FastText	pos1k_male	pos1k_female	0.1214	$< 10^{-2}$	0.2633
	pos5k_male	pos5k_female	0.2895	$< 10^{-2}$	0.1402
	pos10k_male	pos10k_female	0.3602	0.1062	0.0783
GloVe	pos1k_male	pos1k_female	0.1732	$< 10^{-2}$	0.0980
	pos5k_male	pos5k_female	0.2603	$< 10^{-2}$	0.1870
	pos10k_male	pos10k_female	0.3744	0.0467	0.2106

Table 4.8: Gender bias experiments to examine stereotypes related to Math and Arts related domains in Word2Vec, FastText, and GloVe models trained on the Pile dataset. Target sets, created using the *POS Filter* approach, and attribute sets (*math*, *arts*) from the IAT studies were used. Reported WEAT effect sizes and RNSB scores, where only WEAT effect sizes ($p\text{-value} < 0.05$) were considered statistically significant.

sizes indicated a clear association between the *male* terms and *career* terms. In the case of GloVe embeddings, we observed a WEAT *effect size* of 0.3902 (with a $p\text{-value} < 10^{-2}$) and RNSB score of 0.0915 for target sets with 1,000 terms. However, GloVe embeddings produced WEAT *effect sizes* that were not statistically significant ($p\text{-value} > 0.05$) for the other two cases of target sets. In general, all the embeddings showed an association between *male* terms and *career* terms, indicating encoded implicit gender biases.

Study 2 We make use of the newly created target sets, aimed at representing gender groups with words from identified word classes, to examine gender stereotypes associated with Math and Arts related domains. We conducted tests for three word embedding models and the produced results are presented in Table 4.8. The Word2Vec embeddings produced WEAT *effect sizes* that increased by 0.15 with each pair of target sets. A similar trend could also be observed in the case of FastText embeddings. However, FastText embeddings produced a WEAT effect size that is not considered statistically significant ($p\text{-value} > 0.05$) for target sets with 10,000 terms.

Overall, we confirmed that all three embeddings showed an implicit association between target sets and attribute sets, for both the IAT studies under consideration. We further inspect the newly created target sets and discuss the implications of the produced results in Section 4.4.

Career VS Family					
Embeddings	Target sets		WEAT		RNSB
			<i>effect size</i>	<i>p-value</i>	
Word2Vec	cluster_male	cluster_female	0.5620	$< 10^{-3}$	0.3720
FastText	cluster_male	cluster_female	0.5212	$< 10^{-2}$	0.2998
GloVe	cluster_male	cluster_female	0.5601	$< 10^{-3}$	0.3531

Table 4.9: Gender bias experiments to examine stereotypes related to Career and Family related roles in Word2Vec, FastText, and GloVe models trained on the Pile dataset. Target sets, created using the *Semantic Word Clustering* approach, and attribute sets (*career*, *family*) from the IAT studies were used. Reported WEAT effect sizes and RNSB scores, where only WEAT effect sizes ($p\text{-value} < 0.05$) were considered statistically significant.

4.3 Semantic Word Clustering

As part of this approach, we aimed to create word-lists with words that are semantically closer to the target attributes (bias concepts like career, and family) that we intend to examine. As described in Section 3.7, we chose the target bias concepts (attribute sets) as the cluster centroids and grouped all the words in the dataset to create semantically similar (closer in terms of cosine similarity) clusters of words. From each of these clusters, we return a list of *male* and *female* associated words, sorted in the descending order of their cosine similarities to their cluster centroids. We expect these word-lists to be semantically closer to the target bias concepts.

We created target sets (*cluster_male* and *cluster_female*), by choosing the 50 top ranked (in terms of their cosine similarity to their respective cluster centroids) *male* and *female*-associated words from each of the clusters. These target sets are then used in evaluating two gender stereotypes in all three word embedding models.

Study 1 In this case, we clustered words in the *Pile* dataset with a combined list of *career* and *family* terms (16 terms) as initial cluster centroids. Words from each of the 16 cluster words were then classified into *male* and *female*-associated words. We then sorted words in each cluster by their cosine similarities to their respective cluster centroids. By choosing the 50 most similar words (to their cluster centroids) from every cluster, we created *cluster_male* and *cluster_female* target sets. All three word embedding models were evaluated using these target sets to examine the career-gender stereotypes. Word2Vec reported a WEAT *effect size* of 0.5620 and an RNSB score of 0.3720, both indicating a strong association between the *male* and *career*

Math VS Arts					
Embeddings	Target sets		WEAT		RNSB
			<i>effect size</i>	<i>p-value</i>	
Word2Vec	cluster_male	cluster_female	0.3593	$< 10^{-3}$	0.0982
FastText	cluster_male	cluster_female	0.4214	$< 10^{-2}$	0.1798
GloVe	cluster_male	cluster_female	0.4320	$< 10^{-3}$	0.1476

Table 4.10: Gender bias experiments to examine stereotypes related to Math and Arts related domains in Word2Vec, FastText, and GloVe models trained on the Pile dataset. Target sets, created using the *Semantic Word Clustering* approach, and attribute sets (*math*, *arts*) from the IAT studies were used. Reported WEAT effect sizes and RNSB scores, where only WEAT effect sizes ($p\text{-value} < 0.05$) were considered statistically significant.

terms and also between *female* and *family* terms. Similarly, Fasttext and GloVe models also indicate a bias towards *male* target sets with the produced WEAT and RNSB scores presented in Table 4.9.

Study 2 Following the same approach as *Study 1*, we created target sets comprising words from each group, where words are clustered with a combined list of *math* and *arts* terms to as initial cluster centroids. Words are then ordered by their cosine similarities, we choose 50 male and female-associated words from each group as entries into the target sets to represent the gender groups. We evaluated all three word embedding models using the newly created target sets and the reported WEAT and RNSB scores, as shown in Table 4.10. From the reported WEAT *effect size* values, all three embedding models indicate an association between target sets and attribute sets, also indicating a bias towards the *male* group. The RNSB scores presented also show a similar trend in bias for the three models.

Overall, the newly created target sets using the *Semantic Word Clustering* approach are used in confirming implicit gender associations in the trained word embeddings. These results are further discussed in Section 4.4.

4.4 Discussion

In this section, we describe the observations made on the newly created target sets and discuss the experiments for measuring gender bias in word embeddings.

We developed three data-driven approaches for creating word-lists to better represent the gender groups (*male* and *female*) based on the statistical infor-

mation of words in the underlying dataset. In the context of our work, three word embedding models (*Word2Vec*, *FastText* and *GloVe*) were trained on the preprocessed *Pile* dataset (refer Section 3.2) consisting of 8,885,798 unique tokens. The word-lists created using our approaches were then used in examining two gender-related stereotypes (Section 3.3) in these word embeddings using WEAT and RNSB metrics.

All three developed approaches rely on a *Gender classifier*, which is used to infer gender labels for words in the dataset. Since we evaluate gender bias in three word embeddings, we trained three classifiers (described in Section 3.4). These classifiers classify a word into one of the three classes (*male*, *female* or *neutral*). From the evaluation results (described in Section 3.4.1) of the three gender classifiers (*w2v_classifier*, *ft_classifier* and *gl_classifier*), it can be noticed that the False Positive Rate for *female* class is particularly higher, indicating that many data instance that actually belongs to *male* or *neutral* classes are incorrectly classified as *female*. It was observed that when these classifiers were used in our approaches to infer gender labels to identify gender-associated words in the dataset, many words that actually belong to *male* or *neutral* groups were wrongly classified as *female*, which is further discussed in the following.

Frequency First Leveraging previously trained gender classifier, we classified the unique words in the *Pile* dataset, to identify groups of *male*, *female* and *neutral* associated words. The *male* and *female* groups were then sorted in the descending order of their word frequency. We then created 3 target sets for *male* and *female* groups, each containing 1,000, 5,000 and 10,000 gender-associated words. This process was performed using all three word embedding types as feature input words and created target sets on the basis of labels inferred from each gender classifier. The target sets created on the basis of *w2v_classifier* were then used to evaluate the *Word2Vec* embeddings. The target sets created on the basis of *ft_classifier* were used to evaluate the *FastText* embeddings and similarly so for *GloVe* embeddings.

Upon manually inspecting the newly created word-lists, we observed that all three classifiers resulted in similar classifications, except for a few words. For example, *male* target sets created on the basis of *FastText* embeddings included words like *dj* and *danger*, however, these words were classified as *neutral* and thus were not included in *male* target sets created on the basis of *Word2Vec* embeddings. Similarly, we found few words (e.g., *winter*, *disaster*) which were part of the created target sets on the basis of *GloVe* embeddings, were found to be classified as *neutral* in the case of *FastText* and *Word2Vec* embeddings. Although there are few differences in the words in the created target sets, most words in *male* and *female* target sets, created on the basis of

one embedding type matched the target sets of other embedding types.

In general, all the target sets created using this approach, included gendered words (*he, her, husband, wife, females etc*), names of people (*john, julia, maria etc*) and other gender-associated words in respective target sets, showing potential to be a reliable approach for representing a particular gender. Having said that, target sets created using this approach suffer heavily from the inclusion of many words that are gender-neutral in nature. For example, words like *to, it, was* etc were found in *male* target sets and words like *the, of, and* etc were found in *female* target sets. We also found words that were either abbreviations or random characters together (*mgs, kgzhs* etc) and also words with underscores (*gamma_, yu_, omega_* etc), which could lead to poor representation of gender groups. These findings also suggest that a filtering process of words is additionally required in order to exclude such terms.

Furthermore, the created target sets (*top1k_male/female, top5k_male/female* and *top10k_male/female*) were used to examine two gender stereotypes (described in Section 3.3) in word embedding using WEAT and RNSB metrics. For the *Career vs Family* study, the metric results are presented in Table 4.4. Word2Vec resulted in positive WEAT effect sizes for all three cases, indicating an association between *male* associated terms and *career* related terms. FastText embeddings also showed a similar trend for all three cases but resulted in WEAT *effect size* with *p-value* > 0.05 for *top10k_male/female* target sets. GloVe embeddings also showed bias towards *male* terms in the case of *top1k_male/female* target sets. However, resulted in WEAT *effect sizes* with *p-value* > 0.05 for the other two cases. RNSB scores for all three word embeddings also confirmed a bias in the word embeddings.

For the *Math vs Arts* study, the metric results are presented in Table 4.4. Individual results for each of the word embeddings are discussed in Section 4.1.

POS Filter This approach proposes another data-driven method for creating word-lists, by analyzing the POS classes of words. Following the steps described in Section 3.6, we created target sets for each word embedding type, consistent with POS classes (word classes) that are generally used to describe particular gender groups. We find the most frequently occurring gendered words and gender-associated words in each of the identified word classes (described in Section 3.6). These words are then used as entries into the target sets to represent each gender group. A sample list of words that are used as entries in the target sets is presented in Table 4.6.

Manually inspecting the individual terms in the list, we noticed issues similar to that found in the *Frequency First* approach. A noticeable difference is that filtering by POS classes like *NOUN* (Nouns), *PROPN* (Proper Nouns)

and *PRON* (Pronouns) increases the number of gendered words in the target sets, which is desirable especially when we are trying to represent gender groups. However, when poorly performing gender classifiers are employed, it could lead to the inclusion of gender-neutral words in the target sets that might misrepresent a gender group. Particularly when including terms from *VERB* (Verbs) and *ADJ* (Adjectives) POS classes, extra care needs to be taken as words that are not generally associated with a particular group could lead to skewed representation of gender groups. This approach needs more investigation in terms of including words from POS classes other than identified classes in our work.

When testing for traditional job role stereotypes, we observed that word2vec embeddings produced statistically significant WEAT *effect sizes* in all three cases, indicating a bias towards *male* group for the *career* and *family* categories. The RNSB scores for word2vec embeddings showed very low negative sentiment towards the target set terms in all three cases. Although there is a significant difference in the measured WEAT *effect sizes* for FastText embeddings with respect to its Baseline test results, the WEAT effect sizes indicated a clear association between the *male* terms and *career* terms. In the case of GloVe embeddings, we observed a WEAT *effect size* of 0.3902 (with a $p\text{-value} < 10^{-2}$) and RNSB score of 0.0915 for target sets with 1,000 terms. However, GloVe embeddings produced WEAT *effect sizes* that were not statistically significant ($p\text{-value} > 0.05$) for the other two cases of target sets. In general, all the embeddings showed an association between *male* terms and *career* terms, indicating encoded implicit gender biases.

We also conducted tests for three word embedding models, to examine implicit stereotypes associated with *Math* and *Arts* related academic domains. All the test results are presented in Table 4.8. The Word2Vec embeddings produced WEAT *effect sizes* that increased by 0.15 with each pair of target sets. A similar trend could also be observed in the case of FastText embeddings. However, FastText embeddings produced a WEAT effect size that is not considered statistically significant ($p\text{-value} > 0.05$) for target sets with 10,000 terms. GloVe embeddings produced positive WEAT *effect sizes* in all three cases and the reported RNSB scores confirmed a similar trend in the encoded bias against *female* terms.

Semantic Word Clustering With an aim to increase semantically closer terms with the associated target bias concepts, we proposed a bottom-up clustering based approach. Here the target bias concepts are chosen as the cluster centroids and words in the dataset are clustered into semantically cohesive groups. From each of the groups, we selected gender-associated words as entries into the target sets.

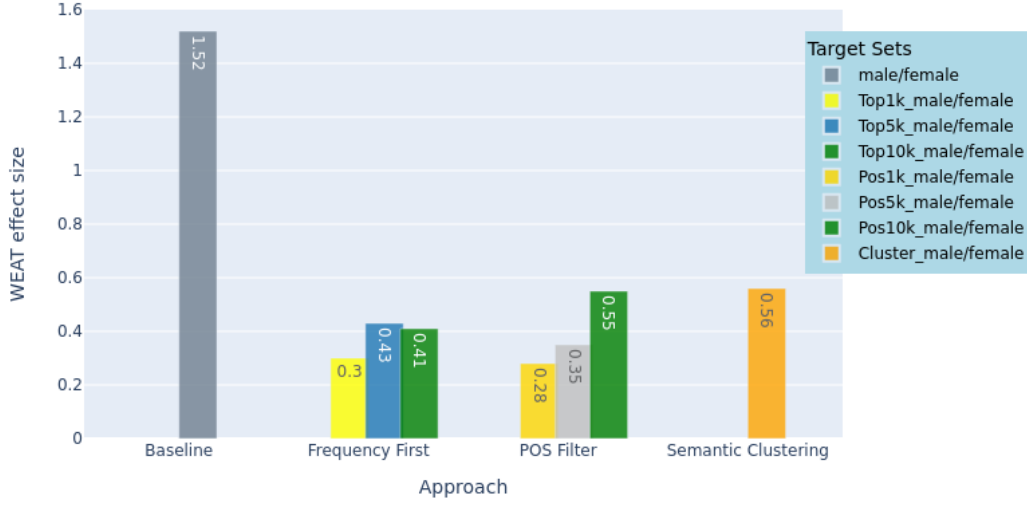


Figure 4.1: Comparison of Word2Vec gender biases reported in terms of WEAT effect sizes for *Career vs Family* study, using all target sets created as part of this thesis work.

Manually inspecting the word-lists, indicated that the number of traditionally gendered words was reduced. In the case when examining job roles related stereotypes associated with gender, we noticed that only the only gendered words included were *he*, *she*, *men* and *boys*. But however, by choosing the 50 top ranked (in terms of word cosine similarity to respective cluster centroids) the semantic cohesion of lists is expected to increase. All three word embedding models were evaluated using these target sets to examine the career-gender stereotypes. Word2Vec reported a WEAT *effect size* of 0.5620 and an RNSB score of 0.3720, both indicating a strong association between the *male* and *career* terms and also between *female* and *family* terms. Fasttext and GloVe models also indicate a bias towards *male* target sets. WEAT and RNSB scores for all the tests performed are presented in Table 4.9. When the newly created target sets were used to examine the stereotype associated with *math* and *arts*-related domains, the reported WEAT *effect size* values for all three embedding models indicate an association between target sets and attribute sets, indicating a bias towards the *male* group. The RNSB scores presented also show a similar trend in bias for the three models. WEAT and RNSB scores for all the tests performed for the second study are presented in Table 4.9.

Figures 4.1 4.2, provide a comparison of Word2Vec word embedding biases reported in terms of WEAT *effect sizes* examining two gender stereotypes. Overall, all three approaches provide a good starting point as data-driven

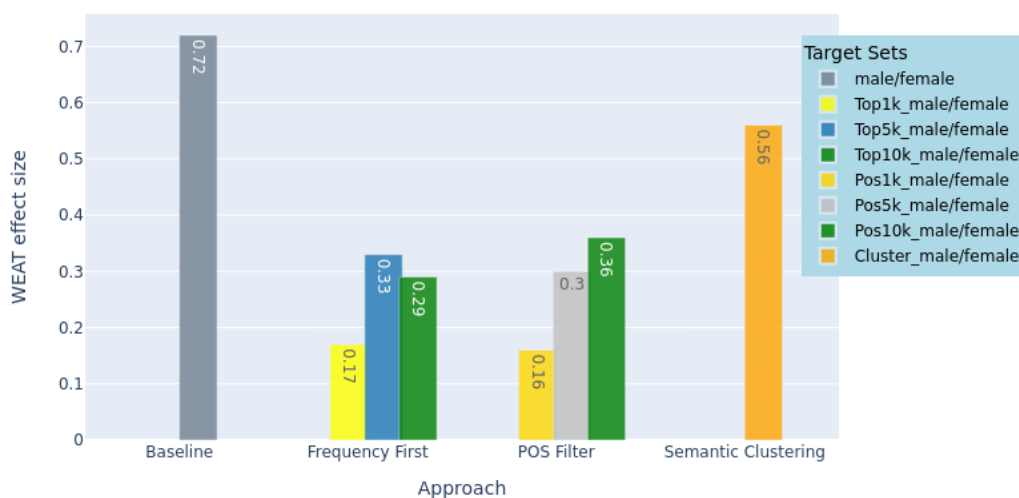


Figure 4.2: Comparison of Word2Vec gender biases reported in terms of WEAT effect sizes for *Math vs Arts* study, using all target sets created as part of this thesis work.

methodologies for word-list creation. However each approach presents a different challenge, that needs to be addressed in order to better represent the social groups. The gender classifier which is primary to all three approaches needs further performance tuning in order to optimize the process of identifying gender-associated words. With all three approaches, a general observation is that the word-lists need a round of manual filtering to exclude terms that could otherwise lead to skewed or misrepresentation of gender groups.

Chapter 5

Conclusion

In the following, the experiments and the resulting findings of this work are summarized with respect to the research questions before addressing their limitations and possible starting points for future work.

5.1 Summary

Word embeddings are a set of Natural Language Processing(NLP) algorithms for mapping words into numerical vectors [Papakyriakopoulos et al., 2020]. Word embeddings trained on massive amounts of human-produced text would result in word embeddings that not only capture human semantics but also eventually end up encoding various kinds of negative societal biases and stereotypes prevalent in the human-produced text. Thus measuring such human-like social biases is key for better understanding and addressing unfairness in word embeddings. This is often done via statistical measures (or bias metrics) such as Word Embedding Association Test [Caliskan et al., 2016] and Relative Negative Sentiment Bias [Sweeney and Najafian, 2019] that stem from Implicit Association Tests(IAT) [Greenwald et al., 1998]. These metrics quantify how word embeddings associate a certain social group (*male*, *female* etc) with some bias-conveying concept(*career*, *family*-related words) and how that differs for another group. Here each of the concepts is described by a list of words.

Bias metrics relying on word-lists are not very well evaluated and show significant limitations in precisely measuring social biases [Garg et al., 2018, Hoyle et al., 2019]. Word lists as of now are sometimes crowd-sourced, sometimes hand-selected by researchers and sometimes drawn from prior work in the social sciences. These word-lists come pre-validated and do not work well in the case of newly found domains of data. Based on work by the authors of [Antoniak and Mimno, 2021, Caliskan et al., 2022] we identified factors of word-lists like word frequency (WEAT tests require the paired target sets to occur at similar frequencies for more accurate measurements), their associated

POS (given that "men" are perceived as more active than "women" [Hsu et al., 2022]), it is possible that male-associated words will be more likely to be verbs and female-associated words to be adjectives and adverbs), and semantic cohesiveness between words that could have an influence on the representation of social groups. Based on these identified factors we proposed 3 data-driven methods *Frequency First*, *Pos Filter* and *Semantic Word Clustering*, to create word-lists for measuring gender bias in text corpora that the word embeddings are trained on.

In the context of our work, we train three word embedding types (Word2Vec, FastText and GloVe) on a large English language dataset called the *Pile*. It comprises 22 subsets of text data from diverse sources (e.g., Books, Research Papers, Youtube Subtitles, Court proceedings etc). These custom-trained word embeddings are then evaluated using bias metrics like *WEAT* and *RNSB* to examine two gender stereotypes such as 1) *Career vs Family* IAT (stereotype: *men* are more associated with professional job roles while *women* are more associated with traditional family roles) and 2) *Math vs Arts* Gender IAT (stereotype: *female* terms are more associated with arts than mathematics related academic domains, compared to *male* terms) [Greenwald et al., 1998]. We also train a *gender classifier* which is used to infer gender labels and is considered an integral part of the developed approaches. Since we aimed to evaluate three word embeddings, we trained 3 gender classifiers on a combined dataset (comprising a total of 44,465 words gathered from *gendered words* subset of MDGender dataset [Dinan et al., 2020] and a list of gender-neutral terms from WordNet [Fellbaum, (1998, ed.)]). For each gender classifier, we represented words with each type of Word embedding type. In choosing the best-performing gender classifier for each embedding type, we trained four Dummy classifiers (provided by *sklearn*, Dummy Classifiers provide four basic strategies for classification tasks) as a Baseline and also more sophisticated models like SVM and Random Forrest. For Word2Vec embeddings, SVM was chosen as the best classifier (*w2v_classifier*). In the case of FastText embeddings also SVM performed the best (*ft_classifier*). For GloVe embeddings, the Random Forest model was found to be the best-performing one (*gl_classifier*).

As part of the *Frequency First* approach, we aim to represent the two gender-based social groups (*male* and *female*) with the most frequently occurring gender-associated words in the dataset. To create word-lists using this approach, we performed three steps. The first step involved inferring gender labels for words in the dataset. For this, we leveraged the best performing gender classifier for an embedding type. We represented every word in the dataset with word embedding as feature input. This is then used to infer gender labels, and classify words into *male*, *female* and *neutral* word groups. From each of these word groups, we then choose 1,000 (*top1k_male*, *top1k_female*), 5,000

(*top5k_male*, *top5k_female*) and 10,000 (*top10k_male*, *top10k_female*) most frequently occurring words from each group as entries into the target sets.

Using these newly created target sets for each word embedding, we evaluate the embeddings two gender stereotypes. The WEAT effect sizes and RNSB scores for career vs family study are provided in Table 4.4. In general, all three word embeddings resulted in positive WEAT effect sizes, indicating an association between *male* and *career* terms and also between *female* and *family* terms. The WEAT effect sizes and RNSB scores for *math* vs *arts* study are provided in Table 4.4.

As part of the *POS Filter*, we aim to represent both the gender groups with words from all the relevant POS classes (*PROPN*, *NOUN*, *PRON*, *ADJ* and *VERB*). This approach is also achieved in three steps. Firstly, we tag all the words in the *Pile* dataset into 17 universal POS tags. From this, we filter out all the words to keep only words from the identified relevant classes. For each class of words, we classify words into *male*, *female* and *neutral* word groups by leveraging respective best performing gender classifiers for each embedding type. Now from the gender associated groups, we then choose 1,000 (*pos1k_male*, *pos1k_female*), 5,000 (*pos5k_male*, *pos5k_female*) and 10,000 (*pos10k_male*, *pos10k_female*) most frequently occurring words from each group as entries into the target sets. Similar to the first approach, we evaluate three word embeddings to examine two gender stereotypes. All the results are provided in Tables 4.7 and 4.8.

The third approach for creating word-lists is *Semantic Word Clustering*, where we aim to represent gender groups with semantically cohesive word-lists with the associated target bias concept. We follow a bottom-up approach, where we cluster all the words in the dataset by choosing the target bias concepts as initial cluster centroids. Then from each cluster, we group all the gender-associated words by leveraging the gender classifiers. We then choose 50 most similar words (to their centroids in terms of cosine similarity) from each gender group as entries into the target sets. This way we expect the lists to be semantically cohesive. The results of evaluation using these target sets are provided in Table 4.9 and Table 4.10. Refer to Figure 4.1 for the word2vec bias results for *career* vs *family* study, using target sets created by all three proposed methods.

With these three data-driven approaches, we attempt to systematically generate target sets by taking into account the underlying dataset and also that could better represent the intended social groups.

5.1.1 Limitations

Limitation 1: Poor Gender Classifier performance: All three gender classifiers (*w2v_classifier*, *ft_classifier* and *gl_classifier*) results in high number of misclassifications. As described in Section 2.2.1.1, the trained classifiers showed a False Positive Rate of 0.45 for Female classes, indicating that 45% of words that actually belong to *male* or *neutral* classes are classified as *female* classes.

Limitation 2: Created target sets are noisy and require further inspection: Target sets created using all three developed approaches are noisy, especially in the case of target sets with 5,000 and 10,000 terms. These target sets need further inspection and even a round of term filtering for them to better represent the intended social groups.

5.2 Future Work

In the context of our work, we focused on the two largest gender social groups *male* and *female*. However in order to be able to represent more diverse gender groups, extending our work to other representations of gender, sex, and intersectional categories would be interesting. We evaluated our approaches on the Word embeddings trained on the *Pile* dataset. However, using these methods on word embeddings trained different datasets of varying sizes would lead to more interesting insights into the developed methods.

Apart from the identified lexical factors that could influence the bias measurement, Future work could also investigate like number of words in the word-lists or ordering of words in the word-list to identify more factors that could influence the bias measurements. We believe that the choice of word-lists is very crucial for measuring bias in word embeddings and our work is just a step towards improving that choice.

Appendix A

Target Sets

A.1 Frequency First

A full list of words in *top1k_male* and *top1k_female* target sets, which were classified using the Word2Vec based gender classifier (*w2v_classifier*) is given here,

top1k_male [to, it, was, you, this, be, not, have, he, can, his, has, there, will, what, also, would, up, who, me, new, than, ref, fig, just, them, how, data, could, said, him, using, now, people, court, through, frac, re, should, did, us, before, case, work, high, years, made, category, much, found, let, based, year, long, without, information, al, must, states, et, following, around, ll, th, going, too, last, given, line, something, cr, might, com, analysis, never, really, text, sec, help, process, big, label, further, next, got, id, evidence, mathbb, aligned, general, mean, things, low, law, test, light, didn, city, open, give, though, away, lambda, file, trial, studies, gamma, research, de, later, similar, district, university, mathbf, cases, along, http, able, true, question, game, national, already, done, rate, course, error, early, men, far, yet, came, include, team, bib, beta, levels, start, response, lot, yes, systems, took, issue, mr, app, room, nothing, doing, provided, conditions, told, cannot, protein, www, god, actually, others, went, terms, performed, war, effects, solution, ii, methods, money, program, anything, asked, art, points, air, string, says, key, red, review, close, content, short, obtained, st, phi, motion, death, project, performance, makes, john, blood, york, future, film, showed, getting, prior, lines, community, models, int, version, simple, report, won, phase, cancer, previous, length, words, added, media, hours, south, sample, rule, via, decision, probably, document, strong, org, original, car, minutes, eyes, wanted, received, theory, works, factors, growth, county, problems, circuit, north, complex, tilde, min, father, gene, music, saw, behind, near, significantly, log, search, involved, building,

knew, plan, looked, written, cross, functions, lead, recent, follow, account, dr, samples, genes, cd, images, video, generally, events, issues, properties, width, nature, limited, div, ct, million, questions, parameters, clinical, department, answer, training, ml, tried, cir, began, cal, address, claims, knowledge, types, late, thanks, https, press, pt, coming, includes, isn, wasn, supplementary, previously, judge, technology, west, thank, science, felt, talk, followed, etc, relationship, management, psi, xi, gave, direct, ed, appeal, release, fine, land, web, agreement, yeah, inc, android, published, soon, games, recently, numbers, friends, road, software, fixed, statement, ago, hands, percent, players, heard, devices, alone, price, street, prime, theorem, false, box, var, computer, context, induced, measured, phys, bf, proof, impact, proposed, link, operatorname, pre, opinion, require, sound, college, message, authors, produced, phone, discussion, brought, wide, parts, gives, ways, hospital, proteins, attention, u_, son, java, drive, deal, myself, continue, sum, looks, sent, rules, wrote, int_, generated, experiments, failed, meet, requires, london, saying, links, earlier, books, rates, g_, deep, summary, decided, vs, bank, lack, closed, hear, findings, park, reasons, google, dead, china, net, applications, quad, okay, ready, random, couldn, supported, maximum, specifically, sub, procedure, stated, files, fields, iii, basic, ca, companies, hour, fast, score, processes, hearing, items, comparison, variables, ones, database, released, activities, panel, reduce, indicated, seemed, starting, round, countries, il, met, concentration, quickly, immediately, calculated, watch, rightrightarrow, david, fall, steps, facts, suggest, varphi, max, noted, tree, success, optical, solutions, library, playing, radio, records, meeting, conclusion, safety, mentioned, quantum, brown, improve, reduction, screen, talking, river, combination, picture, wouldn, rev, digital, testing, structures, match, track, solid, win, comments, died, options, james, mark, trust, washington, lord, suggested, reached, update, guy, actions, sea, administration, bill, tools, span, ph, sorry, conducted, indicate, justice, statistical, void, truth, sensors, denied, supreme, profile, shot, li, glass, race, pat, mid, certainly, kids, agree, stuff, correlation, tests, demonstrated, serious, plants, default, discussed, san, council, testimony, chief, noise, raised, introduced, php, techniques, explained, grant, units, improved, sir, pp, otimes, ideas, appeals, opening, prepared, script, opened, tv, finite, neither, circumstances, tr, programs, credit, absence, requirements, hey, boy, killed, paul, collected, george, names, operations, analyses, classes, ok, ad, analyzed, stock, hall, husband, comment, super, brother, smith, films, texas, en, controls, gonna, minute, michael, tested, guys, contained, chemical, package, affect, eventually, contain, measurements, offered, foundation, vert, rise, congress, haven, sales, br, movie, journal, arms, visual, miles, dose, measurement, characters, relief, equations, suggests, william, drugs, waiting, sequences, laws, patterns, confirmed, thomas, indicates, plans, td, salt, slow, curve, ordered, shared, errors, pcr, con-

ference, robert, cool, projects, networks, nuclear, responses, customers, completed, px, survey, subsequent, brief, environmental, doubt, efficiency, schools, investigation, ci, battle, ran, kg, iv, advanced, campaign, quick, pages, television, stopped, capital, agreed, album, efforts, wind, detailed, technical, sin, approved, cit, ahead, genetic, demand, math, courts, pa, granted, port, runs, van, criteria, ch, standards, hotel, pdf, articles, route, evaluation, orders, lake, aren, accuracy, documents, maps, imaging, faith, carbon, compare, modified, protocol, underlying, nd, m, latest, plot, facebook, relations, hundred, concentrations, nation, helped, positions, implies, percentage, serum, se, anyway, allowing, compounds, discuss, speech, peace, mir, statements, conclude, signaling, procedures, jesus, teams, iron, ill, stone, forth, hill, obviously, discovered, biological, versus, interview, scope, rd, efficient, assay, approaches, fell, grand, wt, decisions, em, ar, unable, fuel, tag, mine, sensitive, keeping, medicine, bridge, proposition, gate, buffer, scores, asking, reverse, announced, instructions, implementation, outcomes, manuscript, radiation, apparently, watching, advice, jobs, valid, pieces, def, examination, limitations, seconds, emission, settings, dynamic, sides, letters, trip, receptor, js, charles, ended, delay, widehat, charges, evaluated, trials, peter, lee, fees, custom, examined, integrated, johnson, wood, trouble, broad, consideration, oral, israel, laser, edit, applying, arrived, arguments, dad, informed, mission, officials, improvement, differential, discovery, median, advance, interval, richard, goals, concerns, boys, thermal, originally, branch, funding, definitely, involving, align, walls, museum, helps, spatial, resulted, challenges, aspects, estimates, raise, chicago, demonstrate, cos, putting, leaders, telling, laboratory, assistance, communities, cash, americans, investment, font, theta_, submitted, historical, bars, physics, cars, rs, favor, papers, exp, cloud, prices, concerning, cc, info, sp, grid, concluded, significance, unfortunately, vision, signs, distributed, conclusions, tour, jones, reducing, unlike, billion, yield, passing, guidelines, feedback, practices, python, api, forest, genome, statistics, funds, tissues, strategies, argued, posted, flight, equiv, categories, bond, mountain, ps, conversation, investigated, proceedings, sky, engineering, valley, spectra, ubuntu, thousands, policies, aside, camp, enhanced, raw, preparation, db, henry, martin, bay, jack, os, obama, amounts, degrees, climate, supp, hello, parking, thick, supra, cf, blocks, bb, edition, practical, dx, songs, steel, monitoring, closely, cards, fans, glucose, technologies, vol, console, messages, marked, tex, adopted, answers, enable, smart, principles, hadn, paris, smile, posts, los, logic, bp, buildings, src, regression, strains, des, farm, surrounding, dropped, communications, columns]

top1k_female [the, of, and, that, for, with, on, by, are, at, from, we, were, all, they, had, my, their, her, time, been, she, about, these, our, type, some, its, any, after, see, here, both, while, however, three, table, day, since, still, life,

world, mathcal, results, value, times, ve, why, show, alpha, little, say, including, again, home, power, above, few, free, pone, fact, support, until, family, days, several, four, delta, area, energy, once, although, values, total, available, real, house, space, full, mu, children, current, usepackage, love, pm, read, pi, women, post, main, note, together, list, making, mathrm, range, view, omega, five, site, x_, sigma, development, history, book, addition, matter, ever, below, night, various, changes, positive, additional, shows, final, taken, interest, almost, today, week, period, series, months, services, a_, mass, clear, please, access, else, region, partial, oh, mm, infty, finally, past, respectively, leq, included, everything, multiple, story, taking, distribution, live, sum_, event, needed, special, m_, started, t_, tau, cost, complete, mother, heart, season, c_, scale, features, products, provides, rho, memory, materials, natural, p_, news, presented, effective, green, cdot, title, woman, central, index, article, online, le, references, nu, background, cm, moment, sqrt, march, v_, initial, la, blue, f_, areas, weeks, month, everyone, perhaps, june, hope, r_, external, mg, indeed, global, entire, dna, share, k_, reading, acid, n_, overline, clearly, description, pretty, stay, town, april, july, purpose, dark, led, overall, january, ms, morning, despite, chapter, america, linear, moreover, offer, notice, september, volume, october, except, details, sites, collection, december, contains, relevant, none, unique, eta, critical, august, ell, legal, b_, wife, november, html, financial, showing, channel, dot, uk, regarding, regions, d_, epsilon, couple, interesting, animals, parents, h_, earth, sex, ten, fully, nice, email, happened, seven, girl, varepsilon, daily, actual, february, costs, sources, sim, gone, sun, resources, usa, visit, happy, windows, california, giving, perfect, summer, plus, lives, hair, female, introduction, continued, internet, chance, kept, providing, eight, experimental, notes, ensuremath, beginning, reports, z_, opportunity, benefits, places, fun, molecules, boldsymbol, worth, stars, minimal, resolution, cycle, safe, blog, nm, examples, alternative, revealed, molecular, ray, i_, extra, beautiful, missing, q_, novel, spring, tab, song, l_, island, gold, india, y_, continuous, measures, recorded, dots, spectrum, chi, miss, widetilde, families, effort, lambda_, europe, peak, germany, extended, fair, sections, england, rna, adding, considering, fresh, figures, offers, stories, ijms, angle, occurred, gamma_, daughter, ij, delta_, w_, photo, enjoy, freedom, kappa, canada, girls, france, sigma_, mrs, planning, sd, importance, seeing, omega_, exact, soft, ge, herself, village, views, intensity, circ, delivery, na, observations, reality, joint, excellent, session, unknown, nine, alpha_, japan, supporting, detail, conflict, ab, recovery, listed, depth, additionally, trees, strongly, mainly, phi_, numerous, coverage, apple, northern, australia, progress, sister, sensitivity, twice, zone, limits, confidence, km, circle, sunday, winter, breast, secret, annual, pc, border, stability, hi, apart, loved, evening, africa, spirit, threshold, ma, friday, observation, relationships, duration, royal, appearance, mathsf,

imagine, initially, absolute, mary, ast, florida, yellow, dc, crystal, warm, zeta, mom, lady, welcome, marriage, accurate, spaces, truly, thoughts, assuming, monday, rose, cities, beach, coffee, compound, sugar, structural, lots, curves, channels, garden, pictures, matters, hidden, saturday, dream, massive, amazing, weather, tables, sweet, nevertheless, absolutely, un, bodies, qqquad, silver, radius, mu_, ultimately, rho_, wine, coast, reviews, universe, locations, dinner, spectral, noticed, virginia, sector, bright, regional, fashion, ln, experiences, episode, abstract, mrna, contents, approx, clusters, sharing, solar, fe, photos, somewhere, supports, integration, universal, besides, momentum, numerical, facilities, wonderful, theoretical, russia, christmas, helpful, carefully, breaking, thursday, dogs, weekend, quiet, si, tea, tuesday, competing, interior, ns, angular, extensive, ijerph, schedule, coefficients, wednesday, updated, gift, moments, formal, planned, dance, afternoon, colors, orange, reads, journey, queen, tomorrow, rooms, privacy, houses, theme, kinds, mexico, ul, complexity, periods, fruit, galaxy, crucial, tau_, homes, era, happening, perfectly, treatments, surprise, directions, italy, nearby, beta_, er, tonight, statistically, bringing, ann, crisis, snow, dear, choices, bc, da, rain, partially, foods, neighborhood, festival, cl, adequate, clothes, videos, bulk, valuable, busy, availability, forum, precise, cs, beauty, ocean, carolina, reaching, angeles, lists, g, roots, inclusion, au, yields, pr, comprehensive, moon, dress, victory, colour, meanwhile, tips, bell, asia, historic, eu, uncertainty, pacific, province, females, segment, wedding, launched, continuing, temple, explicit, feelings, eco, xi_, underline, utility, glad, sodium, sm, leaf, ultimate, awareness, elsewhere, exclusive, invasion, nutrients, forever, locally, pregnancy, seeds, uv, appreciate, flowers, shopping, santa, np, everywhere, sessions, entertainment, stellar, fitting, panels, spain, dates, bigg, ongoing, intervals, worldwide, separately, trans, stayed, eggs, monthly, twelve, enjoyed, fiction, dreams, bold, amazon, mb, est, interestingly, pleasure, jpg, beside, weekly, holiday, atomic, entries, effectiveness, independence, grace, absent, gallery, su, featured, awesome, ending, comfort, visitors, memories, islands, childhood, lips, lifetime, commitment, genuine, wealth, ease, theatre, joy, importantly, georgia, kim, lovely, overnight, deeply, ic, saving, mi, strictly, consistently, frozen, mn, lipid, pink, guests, raising, integrity, campus, birthday, branches, programme, circles, darkness, slope, vital, rings, cholesterol, sole, visits, inverse, ec, um, equality, expectations, tears, exciting, simeq, entropy, abc, trends, seasons, pregnant, satisfying, ce, hd, mini, eta_, intense, epsilon_, mutual, bi, gb, dd, reveals, perp, iphone, vitamin, overview, micro, tbl, jane, elizabeth, elementary, chi_, genomic, exclusion, passion, ranges, visiting, termination, turkey, cache, grass, innovation, viewing, tournament, scenes, es, specificity, lc, honey, drama, ai, varphi_, episodes, ba, igg, juice, unity, partly, arc, angles, clothing, notably, dp, remarkable, nonetheless, archive, acceptance, democracy, ignored, sarah, spots, gev, brazil, bibr,

analytical, displaystyle, mothers, maternal, convex, consistency, pgen, olive, dramatic, forgotten, indiana, singing, adventure, kb, wow, prayer, geometric, triangle, nu_, fantastic, staying, constants, anna, exhibits, savings, pride, nights, meaningful, gp, expectation, listing, contest, nc, ruby, incredible, horizon, photographs, diamond, vegetables, hollywood, restaurants, approximate, reception, ladies, beans, clouds, possibilities, wisdom, cu, archives, alliance, kiss, theater, empirical, ordering, participating, fantasy, fm, unnecessary, unexpected, drops, flexibility, voices, hearts, lessons, highlight, desert, purely, nutrition, gc, atp, victoria, downtown, collections, xx, disaster, apj, odot, extraordinary, remedy, analytic, dynamical, hopes, foster, realistic, ni, purple, equity, poetry, presenting, wings, resort, happiness, sn, mhz, stopping, basal, disappeared, mystery, netherlands, optional, eff, narrative, lemon, adoption, ect, maria, amplification, convenience, palm, highlights, essence, naked, anne, alicia, afterwards, vii, planes, gathering, tokyo, villages, dfraction, arrangements, pb, concert, inspiration, antibiotics, divine, indirect, flower, continuum, texts, loving, thy, teaspoon, varepsilon_, supplement, cas, quantification, shops, implementing, transcripts, cats, meals, eleven, suite, drinks]

A.2 POS Filter

A full list of words in *pos1k_male* and *pos1k_female* target sets, which were classified using the Word2Vec based gender classifier (w2v_classifier) is given here,

pos1k_male [new, high, much, last, cr, com, text, process, big, label, further, next, general, low, open, give, similar, able, true, national, early, cannot, protein, key, red, close, content, short, prior, phase, previous, strong, org, original, north, complex, log, involved, plan, looked, cross, recent, follow, div, ct, clinical, ml, cir, cal, late, coming, wasn, supplementary, west, etc, direct, prime, theorem, false, induced, bf, operatorname, require, wide, u_, java, deal, sum, london, summary, vs, dead, china, okay, ready, sub, basic, seemed, il, max, optical, digital, mark, reached, update, span, indicate, statistical, serious, grant, pp, script, tv, tr, paul, ok, comment, super, smith, chemical, affect, haven, visual, slow, cool, nuclear, px, subsequent, environmental, iv, quick, wind, detailed, technical, sin, genetic, van, pdf, route, aren, faith, underlying, nd, latest, hundred, se, biological, efficient, grand, em, unable, sensitive, medicine, valid, def, dynamic, js, peter, integrated, broad, oral, arrived, differential, median, advance, interval, thermal, branch, align, spatial, laboratory, americans, font, historical, exp, tour, forest, posted, equiv, sky, spectra, ubuntu, camp, raw, martin, os, hello, thick, practical, glucose, vol, console, enable, smart,

smile, los, regression, gray, acute, willing, el, kit, fail, williams, twitter, href, eps, sharp, alive, immediate, magic, audio, microsoft, uh, urban, golden, sql, hr, satisfied, css, ref, fig, data, people, court, frac, re, case, work, years, category, year, information, al, states, ll, th, line, something, analysis, sec, help, id, evidence, mathbb, things, law, test, light, city, file, trial, studies, research, district, university, mathbf, cases, http, question, game, rate, course, error, men, include, team, bib, beta, levels, start, response, lot, yes, systems, issue, mr, app, room, nothing, conditions, www, god, others, terms, war, effects, ii, methods, money, program, anything, art, points, air, review, st, motion, death, project, performance, blood, york, future, film, lines, community, models, version, simple, report, cancer, length, words, media, hours, sample, rule, decision, document, car, minutes, eyes, theory, works, factors, growth, county, problems, circuit, tilde, min, father, gene, music, search, building, functions, account, dr, genes, cd, images, events, issues, properties, nature, questions, parameters, department, training, address, claims, knowledge, types, thanks, press, isn, judge, technology, thank, science, talk, relationship, management, psi, xi, ed, appeal, release, fine, land, agreement, yeah, inc, android, games, numbers, friends, road, software, statement, hands, percent, players, devices, price, street, box, var, computer, context, phys, proof, impact, pre, opinion, sound, college, message, authors, phone, discussion, parts, ways, hospital, attention, son, drive, looks, rules, int_, experiments, meet, links, books, rates, bank, lack, findings, park, reasons, google, net, applications, quad, random, couldn, maximum, procedure, files, fields, companies, hour, fast, score, processes, items, comparison, variables, ones, database, activities, panel, countries, concentration, watch, rightrightarrow, fall, steps, suggest, varphi, tree, success, solutions, library, radio, records, meeting, conclusion, safety, quantum, reduction, screen, river, combination, picture, wouldn, rev, structures, track, solid, comments, options, james, trust, washington, lord, guy, actions, sea, administration, bill, tools, ph, sorry, void, sensors, supreme, profile, shot, glass, race, pat, mid, kids, stuff, correlation, tests, plants, default, san, council, testimony, chief, noise, php, techniques, units, sir, otimes, appeals, circumstances, programs, credit, absence, requirements, george, names, operations, analyses, classes, ad, stock, hall, husband, brother, films, texas, en, controls, gonna, minute, michael, guys, package, measurements, foundation, rise, congress, sales, br, movie, journal, arms, miles, measurement, characters, equations, suggests, drugs, sequences, laws, patterns, thomas, plans, td, salt, errors, conference, robert, projects, responses, customers, survey, brief, doubt, efficiency, schools, investigation, ci, battle, kg, campaign, pages, television, capital, album, efforts, cit, demand, math, courts, port, criteria, ch, standards, hotel, articles, evaluation, orders, accuracy, documents, maps, carbon, compare, protocol, m, plot, facebook, relations, concentrations, nation, positions, implies, percentage, serum, com-

pounds, speech, peace, mir, statements, conclude, procedures, jesus, teams, iron, ill, stone, forth, it, you, he, his, what, who, me, them, him, us, myself, widehat, ourselves, ya, whoever, yo, ym, snapchat, ypt, redhat, wechat, hatcher, wegener, xchat, kutcher, wnd, jehoshaphat, wur, opencart, durocher, chitchat, gotthelf, yakov, meself, nhat, yonhap, rochat, windhorst, herpetiformis, pahat, yuxi, funkyhat, farhat, kohat, relf, kaplinghat, shohat, douthat, ichat, yhat, rhat, whitehat, ybaumy, haghhighipour, wootten, dthat, zhat, yeadon, cthat, celf, osteryoung, diethylether, allhat, hovda, yovani, hexchat, weechat, rthat, ofwhat, huttenlocher, tonthat, libelf, muhat, yáñez, wesnoth, wassenhove, joinchat, winecoff, hihat, wolkoff, schuchat, hipchat, helf, kthat, wijshoff, mahathat, werken, lahat, dhat, whorton, serhat, aelf, vhat, nelf, yft, suchthat, okoshi, hoher, hamacher, objnr, fhat, haghghat, wattsupwiththat, sahat, faghat, alignself, wagstaffe, laubschat, waldfogel, sowhat, createchat, xhat, rocketchat, shaphat, rhothat, lachat, wolhusen, ferhat, azelf, qhat, machat, hilscher, mahat, hosei, yenko, ehat, hoelscher, josaphat, allfather, winde, schat, my_output, yake, mulville, gauchat, merschat, yxj, whicher, bahat, cihat, lelf, qalhat, eelf, wackernagel, schwechat, suthat, sphat, oldhat, wombacher, yöurself, herrscher, gammahat, zelf, josephat, manaen, winickoff, testthat, hovan, wajahat, houdaille, forthat, hiwhat, bötticher, twelf, ldhat, farahat, firechat, yatsen, dehat, yanuk, mself, woodington, hulscher, strawhat, funnylookinhat, wesbanco, somewhat, mach_task_self, camelphat, wohlthat, yebo, opsomer, bouchat, kreuther, beuchat, saysthat, matterwhat, hymself, languagehat, yarno, walseth, surethat, toself, melf, wieghorst, youknowwhat, medhat, torchat, mhat, yesalis, was, be, have, can, has, will, would, could, said, using, should, did, made, found, based, must, et, following, going, given, might, got, aligned, mean, didn, lambda, gamma, done, came, took, doing, provided, told, went, performed, solution, asked, string, says, obtained, makes, john, showed, getting, int, won, added, wanted, received, saw, knew, written, lead, samples, video, width, limited, answer, tried, began, https, includes, felt, followed, gave, web, published, fixed, heard, measured, proposed, link, produced, brought, gives, proteins, continue, sent, wrote, generated, failed, requires, saying, g_, deep, decided, closed, hear, supported, stated, iii, ca, hearing, released, reduce, indicated, starting, met, calculated, david, facts, noted, playing, mentioned, improve, talking, testing, match, win, died, suggested, conducted, justice, truth, denied, agree, demonstrated, discussed, raised, introduced, explained, improved, opening, prepared, opened, hey, killed, collected, analyzed, tested, contained, contain, offered, vert, relief, william, waiting, confirmed, indicates, curve, ordered, shared, pcr, networks, completed, ran, advanced, stopped, agreed, approved, granted, runs, lake, imaging, modified, helped, allowing, discuss, signaling, hill, discovered, scope, assay, fell, ar, keeping, scores, asking, announced, watching, ended, charges, evaluated, examined, consider-

ation, applying, informed, boys, involving, helps, resulted, estimates, raise, chicago, putting, telling, submitted, rs, concerning, concluded, distributed, reducing, passing, argued, investigated, aside, enhanced, henry, parking, dx]

pos1k_female [table, mathcal, ve, little, few, free, several, total, available, real, full, current, read, pi, main, omega, x_, various, positive, additional, final, clear, partial, oh, infty, past, multiple, live, special, m_, complete, rho, natural, effective, green, central, online, nu, initial, r_, external, global, entire, n_, pretty, stay, april, purpose, dark, overall, january, linear, december, unique, critical, legal, financial, d_, couple, interesting, sex, nice, daily, actual, february, sim, sun, usa, happy, perfect, hair, female, internet, experimental, worth, minimal, safe, nm, alternative, molecular, i_, extra, beautiful, q_, gold, india, continuous, chi, miss, peak, germany, fresh, ijms, gamma_, delta_, w_, exact, soft, reality, unknown, ab, depth, numerous, australia, sister, sunday, secret, annual, africa, threshold, ma, mary, yellow, crystal, warm, mom, monday, sugar, structural, saturday, massive, sweet, universe, spectral, regional, abstract, solar, fe, universal, numerical, wonderful, theoretical, helpful, thursday, quiet, interior, ns, angular, extensive, formal, orange, queen, theme, ul, crucial, surprise, beta_, ann, snow, dear, neighborhood, cl, adequate, valuable, busy, precise, ocean, comprehensive, historic, eu, pacific, temple, underline, glad, ultimate, awareness, exclusive, invasion, uv, appreciate, santa, np, stellar, monthly, bold, mb, pleasure, jpg, weekly, atomic, absent, lifetime, genuine, kim, overnight, ic, frozen, vital, sole, inverse, ec, um, simeq, pregnant, hd, eta_, intense, mutual, gb, dd, time, see, day, life, world, results, value, show, home, power, pone, fact, support, family, days, delta, area, energy, values, house, space, mu, children, usepackage, pm, women, post, note, list, range, view, site, development, history, book, addition, matter, night, changes, interest, today, week, period, series, months, services, mass, please, access, region, mm, everything, story, distribution, event, tau, cost, mother, heart, season, c_, scale, features, products, memory, materials, p_, news, title, woman, index, article, le, references, cm, moment, sqrt, march, v_, la, blue, f_, areas, weeks, month, everyone, june, hope, mg, dna, share, k_, acid, overline, description, town, july, ms, morning, chapter, moreover, september, volume, october, details, contains, none, eta, august, b_, wife, november, channel, dot, regions, epsilon, animals, parents, h_, earth, email, girl, costs, sources, resources, visit, windows, california, summer, introduction, chance, notes, ensuremath, reports, opportunity, benefits, places, molecules, boldsymbol, stars, resolution, cycle, blog, examples, ray, spring, tab, song, l_, island, measures, dots, spectrum, widetilde, families, effort, lambda_, fair, sections, england, rna, figures, offers, stories, angle, daughter, photo, freedom, kappa, canada, girls, france, sigma_, sd, importance, ge, village, views, in-

tensity, delivery, observations, joint, excellent, session, alpha_, japan, detail, conflict, recovery, trees, phi_, coverage, apple, progress, sensitivity, zone, limits, confidence, km, circle, winter, pc, border, stability, apart, spirit, friday, relationships, duration, royal, appearance, mathsf, absolute, ast, florida, dc, lady, welcome, marriage, spaces, thoughts, cities, beach, coffee, compound, curves, channels, garden, matters, amazing, tables, un, bodies, qqquad, silver, radius, mu_, rho_, wine, coast, reviews, locations, dinner, virginia, sector, fashion, experiences, contents, clusters, photos, integration, momentum, facilities, russia, christmas, dogs, weekend, si, tea, tuesday, ijerph, schedule, coefficients, wednesday, gift, moments, dance, afternoon, colors, reads, journey, tomorrow, rooms, privacy, houses, kinds, complexity, periods, galaxy, tau_, homes, era, treatments, directions, er, bringing, crisis, choices, bc, da, rain, foods, festival, videos, bulk, availability, forum, cs, beauty, carolina, angeles, lists, roots, inclusion, au, moon, dress, victory, colour, tips, bell, asia, uncertainty, province, females, segment, explicit, feelings, eco, xi_, utility, sodium, sm, leaf, nutrients, pregnancy, seeds, flowers, sessions, entertainment, panels, dates, intervals, worldwide, trans, eggs, fiction, dreams, est, holiday, entries, effectiveness, independence, gallery, su, visitors, memories, islands, childhood, lips, commitment, wealth, joy, georgia, deeply, mi, mn, lipid, pink, integrity, birthday, branches, programme, circles, darkness, rings, cholesterol, visits, equality, expectations, tears, entropy, abc, trends, seasons, ce, mini, epsilon_, bi, reveals, perp, vitamin, overview, tbl, jane, elizabeth, chi_, exclusion, passion, ranges, termination, cache, grass, innovation, tournament, scenes, lc, honey, drama, episodes, we, they, my, their, her, she, our, its, herself, wow, wp, heather, whirlwind, apprself, yumiko, yasukichi, melihat, rahat, joy-purhat, whinchat, yoshimi, yūgao, wzi, yokkaichi, webchat, yugioh, parishat, gthat, fthat, ethat, ithat, qthat, sthat, mamoc, dwelf, jthat, rajarhat, parashat, livechat, rautahat, lthat, seyahat, simchat, beacher, yizkor, orecchio, mthat, exoticlang_chat, ofher, ocapi, ohas, zthat, wthat, oneofthem, hamsher, realself, kwhat, knowthat, mantelshelf, gchat, saidthat, panuhat, worklet, biphat, inthat, swthat, omdahl, yangshuo, redshelf, yumei, silghat, khambhat, sothat, ephat, witfoth, kinghat, youghalsw, ouds, willenbucher, furhat, gwhat, ythat, saywhat, whatwhatwhat, mwhat, morewhat, y_arr, knewthat, wonchi, hold-slock, hirsself, mechat, phiphat, wheelchart, widada, josphat, malahat, xthat, hneither, htself, oushadhi, winechat, ohsaka, balurghat, hubbie, y_train_, knowwhat, yūeh, pchat, jvechat, midshelf, website_chat, moonthat, mehat, maharajahat, htmlvalue, methat, suprabhat, dwhat, ghatghat, wtld, ruchat, chimpchat, bookself, yopa, karmanghat, golaghat, comwhat, dowhat, hwhat, qwhat, tonsxchat, comrelf, kernhat, youthis, wahat, weblisten, rehmwhat, lightconeqhat, wasteth, ywhz, waitwhat, outwardmost, hindiiif, yayā, agchat, kkqt, wemds, digichat, fbchat, splashthat, wollnik, majerhat, wajd, defhat,

mutableself, fwhat, showthat, ofboth, upchat, sabkhat, yogshala, remember-
what, sehat, iself, chakkraphat, medschat, rajghat, wellikoff, seethat, ishat,
yndi, kalighat, teelf, yosumin, wp_cron, oceanfisher, newself, overself, top-
shelf, objself, rrahat, rhhat, rvhat, lughat, bluehat, gariahhat, hat, wiremock-
config, yoreself, sabjkhat, æthat, itwhat, webself, goodwhat, ybalo, whhat,
olio_, chethanbhat, youthat, are, were, had, been, type, times, alpha, say,
including, love, making, mathrm, sigma, shows, taken, a_, leq, included, tak-
ing, needed, started, t_, provides, presented, cdot, reading, led, offer, sites,
collection, relevant, ell, html, showing, uk, regarding, ten, happened, gone,
giving, lives, continued, kept, providing, beginning, z_, fun, revealed, miss-
ing, novel, recorded, europe, extended, adding, considering, occurred, ij, en-
joy, mrs, planning, seeing, omega_, circ, supporting, listed, northern, breast,
hi, loved, evening, observation, imagine, zeta, accurate, assuming, rose, lots,
pictures, hidden, dream, weather, noticed, bright, episode, approx, sharing,
supports, breaking, competing, updated, planned, mexico, fruit, happening,
italy, tonight, clothes, reaching, g, yields, pr, wedding, launched, continuing,
shopping, fitting, spain, bigg, ongoing, stayed, twelve, enjoyed, grace, fea-
tured, ending, comfort, ease, saving, guests, raising, campus, slope, exciting,
satisfying, visiting, viewing, specificity, ai, democracy, ignored, sarah, pgen,
forgotten, singing, staying, gp, listing, ruby, horizon, hollywood, alliance, kiss,
ordering, participating, drops, apj, mhz, stopping, ect, maria, gathering, lov-
ing, quantification, implementing, cats, exploring, iso, highlighted, avoiding,
xy, upcoming, scattered, ethanol, dancing, pleasant, demonstrating, cried, ap-
proaching, extracts, pray, ensuring, spreading, crying, globe, anticipated, at-
lanta, aunt, padding, recipes, hosting, titled, virgin, mirna, observing, recalled,
arena, friendship, streams, compelling, enjoying, secrets, standardized, lasting,
courage, dotted, delivering, metabolites, ghz, touching, circulating]

A.3 Semantic Word Clustering

Stereotype	Cluster Centroids	<i>male</i>	<i>female</i>	<i>neutral</i>
<i>Career vs Family</i>	<i>executive</i>	114339	160077	182942
	<i>management</i>	59,425	83,195	95,080
	<i>professional</i>	72,982	102,175	116,771
	<i>corporation</i>	101,838	142,573	162,941
	<i>salary</i>	61,103	85,544	97,765
	<i>office</i>	60,538	84,753	96,861
	<i>business</i>	70,038	98,053	112,061
	<i>career</i>	14,125	19,775	24,860
	<i>home</i>	1,115,818	1,562,145	1,785,309
	<i>parents</i>	55,537	77,752	88,859
	<i>children</i>	41,344	57,882	66,150
	<i>family</i>	49,083	68,716	64,918
	<i>cousins</i>	40,574	63,837	72,957
	<i>marriage</i>	45,598	63,837	72,957
	<i>wedding</i>	51,686	72,360	82,698
	<i>relatives</i>	45,972	64,361	73,555

Table A.1: A summary of number of words in the *Pile* dataset, clustered into 16 groups with *career* and *family* terms as cluster centroids.

Bibliography

Yaser S. Abu-Mostafa, Malik Magdon-Ismael, and Hsuan-Tien Lin. *Learning From Data*. AMLBook, 2012.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *NAACL 2019, 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59, 2019.

Maria Antoniak and David Mimno. Bad seeds: Evaluating lexical methods for bias measurement. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1889–1904, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.148. URL <https://aclanthology.org/2021.acl-long.148>.

Pablo Badilla, Felipe Bravo-Marquez, and Jorge Pérez. Wefe: The word embeddings fairness evaluation framework. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 430–436. International Joint Conferences on Artificial Intelligence Organization, 7 2020. doi: 10.24963/ijcai.2020/60. URL <https://doi.org/10.24963/ijcai.2020/60>.

Mahzarin R. Banaji and Anthony G Greenwald. Implicit gender stereotyping in judgments of fame. *Journal of personality and social psychology*, 68 2: 181–98, 1995.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information, 2016. URL <http://arxiv.org/abs/1607.04606>. cite arxiv:1607.04606Comment: Accepted to TACL. The two first authors contributed equally.

- 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. 07 2016.
- Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001. ISSN 0885-6125. doi: 10.1023/A:1010933404324. URL <http://dx.doi.org/10.1023/A%3A1010933404324>.
- Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora necessarily contain human biases. *CoRR*, abs/1608.07187, 2016. URL <http://arxiv.org/abs/1608.07187>.
- Aylin Caliskan, Pimparkar Ajay, Tessa Charlesworth, Robert Wolfe, and Mahzarin Banaji. Gender bias in word embeddings: A comprehensive analysis of frequency, syntax, and semantics. 06 2022.
- Erion Çano and Maurizio Morisio. Word embeddings for sentiment analysis: A comprehensive empirical survey. *CoRR*, abs/1902.00753, 2019. URL <http://arxiv.org/abs/1902.00753>.
- Andrés Carvallo and Denis Parra. Comparing word embeddings for document screening based on active learning. In *BIRNDL@SIGIR*, 2019.
- Sapna Cheryan and Hazel Markus. Masculine defaults: Identifying and mitigating hidden cultural biases. *Psychological Review*, 127, 08 2020. doi: 10.1037/rev0000209.
- Stack Exchange Community. Stack exchange. :<https://archive.org/details/stackexchange>, 2022. [Online; accessed 07-Jan-2023].
- Michal Danilák. Langdetect. :<https://github.com/Mimino666/langdetect>, 2021. [Online; accessed 08-Jan-2023].
- Sunipa Dev, Tao Li, Jeff M. Phillips, and Vivek Srikumar. On measuring and mitigating biased inferences of word embeddings. *CoRR*, abs/1908.09369, 2019. URL <http://arxiv.org/abs/1908.09369>.
- Patricia Devine. Stereotypes and prejudice: Their automatic and controlled components. *Journal of Personality and Social Psychology*, 56:5–18, 01 1989. doi: 10.1037//0022-3514.56.1.5.
- Emily Dinan, Angela Fan, Ledell Wu, Jason Weston, Douwe Kiela, and Adina Williams. Multi-dimensional gender bias classification. In *Proceedings*

- of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 314–331, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.23. URL <https://aclanthology.org/2020.emnlp-main.23>.
- István Endrédy and Attila Novák. More effective boilerplate removal - the goldminer algorithm. *Polibits*, 48:79–83, 2013. doi: 10.17562/pb-48-10. URL <https://doi.org/10.17562/pb-48-10>.
- Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. Understanding undesirable word embedding associations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1696–1705, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1166. URL <https://aclanthology.org/P19-1166>.
- Ethan Fast, Binbin Chen, and Michael S. Bernstein. Empath: Understanding topic signals in large-scale text. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI ’16, page 4647–4657, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450333627. doi: 10.1145/2858036.2858535. URL <https://doi.org/10.1145/2858036.2858535>.
- Christiane Fellbaum. Wordnet: An electronic lexical database. *Cambridge, MA: MIT Press.*, (2), (1998, ed.).
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The pile: An 800gb dataset of diverse text for language modeling. *CoRR*, abs/2101.00027, 2021. URL <https://arxiv.org/abs/2101.00027>.
- Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences*, 115(16):E3635–E3644, 2018. doi: 10.1073/pnas.1720347115. URL <https://www.pnas.org/doi/abs/10.1073/pnas.1720347115>.
- Hila Gonen and Yoav Goldberg. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 609–614, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1061. URL <https://aclanthology.org/N19-1061>.

- Anthony G Greenwald, Debbie E. McGhee, and Jordan L. K. Schwartz. Measuring individual differences in implicit cognition: the implicit association test. *Journal of personality and social psychology*, 74 6:1464–80, 1998.
- Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The elements of statistical learning: data mining, inference and prediction*. Springer, 2 edition, 2009. URL <http://www-stat.stanford.edu/~tibs/ElemStatLearn/>.
- M.A. Hearst, S.T. Dumais, E. Osuna, J. Platt, and B. Scholkopf. Support vector machines. *IEEE Intelligent Systems and their Applications*, 13(4): 18–28, 1998. doi: 10.1109/5254.708428.
- Samuel Russell Hodge, Joe W. Burden, Leah E. Robinson, and Robert Anthony Bennett. Theorizing on the stereotyping of black male student-athletes: Issues and implications. *Journal for the Study of Sports and Athletes in Education*, 2:203 – 226, 2008.
- Alexander Miserlis Hoyle, Lawrence Wolf-Sonkin, Hanna Wallach, Isabelle Augenstein, and Ryan Cotterell. Unsupervised discovery of gendered language through latent-variable modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1706–1716, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1167. URL <https://aclanthology.org/P19-1167>.
- Ning Hsu, Daniel A. Newman, and Katie L. Badura. Emotional intelligence and transformational leadership: Meta-analysis and explanatory model of female leadership advantage. *Journal of Intelligence*, 10(4), 2022. ISSN 2079-3200. doi: 10.3390/jintelligence10040104. URL <https://www.mdpi.com/2079-3200/10/4/104>.
- Larry L. Jacoby, Judith Brown, and Jennifer Jasechko. *Becoming famous overnight : Limits on the ability to avoid unconscious influences of the past*. 2004.
- Kenneth Joseph, Wei Wei, and Kathleen M Carley. Girls rule, boys drool: Extracting semantic and affective stereotypes from twitter. In *Proceedings of the 2017 ACM conference on computer supported cooperative work and social computing*, pages 1362–1374, 2017.
- Dan Jurafsky and James H. Martin. *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Pearson Prentice Hall, Upper Saddle River, N.J., 2009. ISBN 9780131873216 0131873210. URL

- http://www.amazon.com/Speech-Language-Processing-2nd-Edition/dp/0131873210/ref=pd_bxgy_b_img_y.
- Sabrina Keene. *Social Bias: Prejudice, Stereotyping, and Discrimination*. The Journal Of Law Enforcement (Vol. 1, No. 3)., 2011.
- Rahul Khanna and Mariette Awad. *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. 04 2015. ISBN 1430259892. doi: 10.1007/978-1-4302-5990-9.
- Bryan Klimt and Yiming Yang. The enron corpus: A new dataset for email classification research. In *Machine Learning: ECML 2004*, volume Volume 3201/2004, pages 217–226. Springer Berlin / Heidelberg, 2004. doi: 10.1.1.61.1645. URL <http://www.springerlink.com/content/q8g7blqvqyxrvap/>.
- Philipp Koehn. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Conference Proceedings: the tenth Machine Translation Summit*, pages 79–86, Phuket, Thailand, 2005. AAMT, AAMT. URL <http://mt-archive.info/MTS-2005-Koehn.pdf>.
- Austin C. Kozlowski, Matt Taddy, and James A. Evans. The geometry of culture: Analyzing the meanings of class through word embeddings. *American Sociological Review*, 84(5):905–949, 2019. doi: 10.1177/0003122419877135. URL <https://doi.org/10.1177/0003122419877135>.
- Amit Mandelbaum and Adi Shalev. Word embeddings and their use in sentence classification tasks. *CoRR*, abs/1610.08229, 2016. URL <http://arxiv.org/abs/1610.08229>.
- Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999. URL <http://nlp.stanford.edu/fsnlp/>.
- Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pages 55–60. The Association for Computer Linguistics, 2014. ISBN 978-1-941643-00-6. URL <http://dblp.uni-trier.de/db/conf/acl/acl2014-d.html#ManningSBFBM14>.
- Mitchell Marcus, Mary Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19:313–330, 07 2002.

- J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon. A PROPOSAL FOR THE DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE. <http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html>, 1955. URL <http://www-formal.stanford.edu/jmc/history/dartmouth/dartmouth.html>.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013a. URL <http://dblp.uni-trier.de/db/journals/corr/corr1301.html#abs-1301-3781>.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013b.
- Tom M Mitchell. *Machine learning*, volume 1. McGraw-hill New York, 1997.
- Usman Naseem, Imran Razzak, Shah Khalid Khan, and Mukesh Prasad. A comprehensive survey on word representation models: From classical to state-of-the-art word representation language models. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 20(5), jun 2021. ISSN 2375-4699. doi: 10.1145/3434237. URL <https://doi.org/10.1145/3434237>.
- Dong Nguyen, Dolf Trieschnigg, A. Seza Doğruöz, Rilana Gravel, Mariët Theune, Theo Meder, and Franciska de Jong. Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1950–1961, Dublin, Ireland, August 2014a. Dublin City University and Association for Computational Linguistics. URL <https://aclanthology.org/C14-1184>.
- Dong Nguyen, Dolf Trieschnigg, A. Seza Doğruöz, Rilana Gravel, Mariët Theune, Theo Meder, and Franciska de Jong. Why gender and age prediction from tweets is hard: Lessons from a crowdsourcing experiment. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1950–1961, Dublin, Ireland, August 2014b. Dublin City University and Association for Computational Linguistics. URL <https://aclanthology.org/C14-1184>.
- Brian Nosek, Mahzarin Banaji, and Anthony Greenwald. Harvesting implicit group attitudes and beliefs from a demonstration web site. *Group Dynamics-*

- theory Research and Practice - GROUP DYN-THEORY RES PRACT*, 6, 03 2002a. doi: 10.1037//1089-2699.6.1.101.
- Brian A. Nosek, Mahzarin R. Banaji, and Anthony G Greenwald. Math male , me female , therefore math me. 2002b.
- Orestis Papakyriakopoulos, Juan Carlos Medina Serrano, and Simon Hegelich. The spread of covid-19 conspiracy theories on social media and the effect of content moderation. 08 2020. doi: 10.37016/mr-2020-034.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830, 2011.
- Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543, 2014.
- Shawn Presser. Books3. ://twitter.com/theshawwn/status/1320282149329784833/, 2020. [Online; accessed 07-Jan-2023].
- Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. When and why are pre-trained word embeddings useful for neural machine translation? 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 529–535, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-2084. URL <https://aclanthology.org/N18-2084>.
- Jack W. Rae, Anna Potapenko, Siddhant M. Jayakumar, and Timothy P. Lillicrap. Compressive transformers for long-range sequence modelling. *CoRR*, abs/1911.05507, 2019. URL <http://arxiv.org/abs/1911.05507>.
- Chelsea A Heuer Rebecca M Puhl. The stigma of obesity: a review and update. *obesity*, 17(5):941–964. 2009.
- Radim Řehůřek and Petr Sojka. Word2vec. :https://radimrehurek.com/gensim/auto_examples/tutorials/run_fasttext.html *fasttext-model*, 2021a. [Online; accessed 14 – Jan – 2023].
- Radim Řehůřek and Petr Sojka. Simple preprocess. :https://radimrehurek.com/gensim/utils.html#gensim.utils.simple_preprocess, 2021b. [Online; accessed 14 – Jan – 2023].

- Radim Řehůřek and Petr Sojka. Word2vec. [:https://radimrehurek.com/gensim/models/word2vec.htmlmodule-gensim.models.word2vec](https://radimrehurek.com/gensim/models/word2vec.htmlmodule-gensim.models.word2vec), 2021c. [Online; accessed 14-Jan-2023].
- Robin Ruede, Markus Müller, Sebastian Stüker, and Alex Waibel. Enhancing backchannel prediction using word embeddings. pages 879–883, 08 2017. doi: 10.21437/Interspeech.2017-1606.
- Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edition, 2010.
- David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. Analysing mathematical reasoning abilities of neural models. *CoRR*, abs/1904.01557, 2019. URL <http://arxiv.org/abs/1904.01557>.
- Teresa Scheid, Eric Wright, David Mechanic, Allan Horwitz, Jerome Wakefield, Mark Schmitz, Owen Whooley, Corey Keyes, Jason Schnittker, Sharon Schwartz, Cheryl Corcoran, Peggy Thoits, Harriet Lefley, Blair Wheaton, Shirin Montazer, Robyn Brown, Gabriele Ciciurkaite, Laura Limonic, Mary Clare Lennon, and Isabelle Beulaygue. A handbook for the study of mental health. 05 2018. doi: 10.1017/9781316471289.
- João Sedoc and Lyle Ungar. The role of protected class word lists in bias identification of contextualized word representations. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 55–61, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-3808. URL <https://aclanthology.org/W19-3808>.
- Noah A. Smith and Jason Eisner. Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL’05)*, pages 354–362, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. doi: 10.3115/1219840.1219884. URL <https://aclanthology.org/P05-1044>.
- Maximilian Spliethöver and Henning Wachsmuth. Bias silhouette analysis: Towards assessing the quality of bias metrics for word embedding models. pages 552–559, 08 2021. doi: 10.24963/ijcai.2021/77.
- Daniel Todd Gilbert Susan Fiske and Gardner Lindzey. *Stereotyping, prejudice, and discrimination*. The handbook of social psychology, pages 357–411. Oxford University Press, Boston and New York., 1998.
- Chris Sweeney and Maryam Najafian. A transparent framework for evaluating unintended demographic bias in word embeddings. pages 1662–1667, 01 2019. doi: 10.18653/v1/P19-1162.

- H. Tajfel and J.C. Turner. The social identity theory of intergroup behavior. In: *Worchel, S. and Austin, W.G., Eds., Psychology of Intergroup Relation, Hall Publishers, Chicago*, 127:7–24, 1986.
- Jörg Tiedemann. Finding alternative translations in a large corpus of movie subtitle. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 3518–3522, Portorož, Slovenia, May 2016. European Language Resources Association (ELRA). URL <https://aclanthology.org/L16-1559>.
- Deborah Trytten, Anna Lowe, and Susan Walden. “asians are good at math. what an awful stereotype” the model minority stereotype’s impact on asian american engineering students. *Journal of Engineering Education*, 101, 07 2012. doi: 10.1002/j.2168-9830.2012.tb00057.x.
- Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William S. Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. Ethical and social risks of harm from language models. *CoRR*, abs/2112.04359, 2021. URL <https://arxiv.org/abs/2112.04359>.
- John E. Williams and Deborah L. Best. Sex stereotypes and trait favorability on the adjective check list. *Educational and Psychological Measurement*, 37 (1):101–110, 1977. doi: 10.1177/001316447703700111. URL <https://doi.org/10.1177/001316447703700111>.
- Robert Wolfe and Aylin Caliskan. Low frequency names exhibit bias and overfitting in contextualizing language models. *CoRR*, abs/2110.00672, 2021. URL <https://arxiv.org/abs/2110.00672>.
- Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2989, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1323. URL <https://aclanthology.org/D17-1323>.