Understanding User Behavior in Social Networks Using Quantified Moral Foundations

By

Pegah Nokhiz

Submitted to the graduate degree program in Department of Electrical Engineering and Computer Science and the Graduate Faculty of the University of Kansas in partial fulfillment of the requirements for the degree of Master of Science.

Prof. Fengjun Li, Chairperson

Committee members

Prof. Bo Luo

Prof. Cuncong Zhong

Date defended: _____

The Thesis Committee for Pegah Nokhiz certifies
that this is the approved version of the following thesis :

Understanding User Behavior in Social Networks Using Quantified Moral Foundations

Prof. Fengjun Li, Chairperson

Date approved: _____

Abstract

Moral inclinations expressed in user-generated content such as online reviews or tweets can provide useful insights to understand users' behavior and activities in social networks, for example, to predict users' rating behavior, perform customer feedback mining, and study users' tendency to spread abusive content on these social platforms. In this work, we want to answer two important research questions. First, *if the moral attributes of social network data can provide additional useful information about users' behavior and how to utilize this information to enhance our understanding*. To answer this question, we used the Moral Foundations Theory and Doc2Vec, a Natural Language Processing technique, to compute the quantified moral loadings of user-generated textual contents in social networks. We used conditional relative frequency and the correlations between the moral foundations as two measures to study the moral break down of the social network data, utilizing a dataset of Yelp reviews and a dataset of tweets on abusive user-generated content. Our findings indicated that these moral features are tightly bound with users' behavior in social networks. The second question we want to answer is *if we can use the quantified moral loadings as new boosting features to improve the differentiation, classification, and prediction of social network activities*. To test our hypothesis, we adopted our new moral features in a multi-class classification approach to distinguish hateful and offensive tweets in a labeled dataset, and compared with the baseline approach that only uses conventional text mining features such as tf-idf features, Part of Speech (PoS) tags, etc. Our findings demonstrated that the moral features improved the performance of the baseline approach in terms of precision, recall, and F-measure.

Acknowledgements

I would like to thank my advisor, Professor Fengjun Li, for mentoring me during my research.

I would like to especially thank my parents, who supported me throughout my life and for all their love and encouragement.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Nowadays, social networks play pivotal roles in our lives. Millions of people are interacting on these social platforms by engaging in various activities such as posting comments, tagging, following, etc. One of the obvious consequences of these phenomena is the huge amount of user-generated content, in particular, textual data which is posted and recorded every day. For instance, Twitter has reached more than 330 million of active users as of the fourth quarter of 2017 [3]. Yelp.com which is known for the users' reviews on businesses, such as restaurants, has 148 million posted reviews by the end of 2017 [5]. This large amount of user-generated content posted daily, contains hidden knowledge about people's opinions, and interactions such as friendships. In addition, users' mandatory and extended profile data, meta data, users' contacts' wall, and private walls are other examples of general types of information extracted from social networks. In more specific examples, we can observe people's tendencies in ranking different businesses, and how they might be misusing social networks to spread abusive content [51].

There are several methods to extract information from social networks' data such as opinion mining which is used to automatically determine human opinion from text. Another approach is scraping for unstructured web data extraction. Sentiment analysis is used to extract subjective

information from social networks' data. Clustering techniques and graph theory-based analytics can be used for unlocking communities and link prediction [8, 33]. Additionally, supervised/semi-supervised/unsupervised methods, as well as classification algorithms can be used for data mining and classification tasks [6]. Finally, text analytics incorporate data/text mining, annotation, and data visualization. For instance, news analytics is an example of capturing the textual attributes of social networks' data to measure novelty, sentiment, and relevance of social media news.

Text mining tries to automatically extract useful knowledge from textual data such as social networks' posts, emails, messages, etc. Some applications of text mining are information extraction, link analysis, and clustering. These applications help unlock the hidden correlations and patterns in textual data that yields information on human behavior [51]. While Natural Language Processing (NLP), is an attempt in the text mining context to extract more latent knowledge from text, e.g., the sentiments, information about the users, their intent in posting a specific message, etc. [57], i.e., NLP is the steps taken to extract useful information from natural language input and/or the use of this information in generating natural language output [10].

1.2 Challenges in Mining Social Networking Data

The performance of text mining methods is not as expected and the textual data on social networks provides a rich source of academic research challenges for the text mining community. The general key challenges are listed below:

Due to the commercial value of the data, social network platforms do not allow comprehensive access to raw data, e.g., Twitter API's limitation for number of accessible tweets. Data cleansing is another key challenge due to the ambiguous, abbreviated, unstructured, and missing information in social networks' textual data [10, 54]. Some social platforms such as Twitter are of very short texts and these short messages might not provide sufficient similarity measures for extracting useful knowledge [26, 54].

Data protection and users' privacy is another challenge [10]. Moreover, there is an abundance

of information on different topics, i.e., each tweet can be associated with multiple labels based on its hashtags, URLs, retweeting status, users' profile, etc. [54]. Data visualization can help extract useful information from abundant natural language data, however, given the magnitude of the data, visualization can face computational issues.

Furthermore, combining different data attributes with a holistic approach can help better understand social networks, e.g., we can combine real-time market information, with textual data, and geo-location tags [10]. However, this data might not be easily available and might suffer from ambiguity. The lack of ground truth in supervised methods can also result in several issues which can be more severe with a holistic approach. Also, user-generated content on social networks is time-sensitive. There might be millions of textual data related to a controversial social event during a specific time-frame [54]. Additionally, many social events are mostly about cultural, behavioral, and moral aspects of the human beings, therefore, each domain needs specific features for text processing rather than a general approach. Quantifying and incorporating these social and moral aspects is a key challenge. Finally, human coders might not be reliable to annotate the data due to their personal biases [25, 102]. This bias can influence supervised methods' outputs, in particular, in terms of moral, social, and behavioral studies performed on social networks' data.

Distinguishing hate speech and offensive language in social networks which is of great interest in the text mining community is an example of these challenges [25]. This differentiation is a difficult task due to the subjective data, lack of training data, and very short texts in social media [25, 26]. These challenges are reflected in a misclassification rate of almost 40% in hate speech detection in a previous study [25].

The following examples can help illustrate the challenges in differentiating hate speech and offensive language.

Hate speech: "Now is the time for the Aryan race 2 stand up and say 'no more'. Before the mong**ls turn the world into a ghetto slum." This sentence is targeting a minority group based on their race.

Offensive language: "D*mb f*cks. Race ba*ing b*tches." This sentence tries to offend others

and is not explicitly targeting a minority group.

Neither: "looking through race colored glasses." This sentence is a comparatively innocent comment which is neither targeting nor offending other users.

As it can be implied from the examples above, the differentiation between hate speech and offensive language is challenging for human coders and automatic NLP techniques due to the subjective definitions and manifestations of these contents in social networks. To be more specific, hate speech is a malicious, biased speech in social networks which targets, degrades, and humiliates victims based on their intrinsic characteristics such as race, gender, sexuality, ethnicity, etc. It can be potentially harmful to people who are members of these groups and has inevitably increased with the fast-growing social interactions in social networks.

However, offensive language is different from hate speech since the offensive words are not used in the same manner as hate speech. In hate speech, the “dangerous” speech, humiliates victims and invites for violence against a minority group [39] while in offensive language the offensive words might be simply part of daily conversations which have a relatively high prevalence on social media or they can be popular slurs among teenagers [99, 101].

It is notable that ‘*’ signs are not part of the original tweets in the previous examples and are inserted by us. All tweets have been slightly changed without changing their original meaning to protect users’ privacy.

1.3 Existing Solutions

Nowadays people’s interactions on social networks are considered a new source that reveals human’s psychological and behavioral patterns. The influences on users’ online activities has been extensively studied in recent years in the communities of computer science, sociology, management and psychology [20, 24, 78, 96]. Besides the textual content of social networks’ data, several variables, such as the counts of upvotes and downvotes, usefulness, coolness, etc., are introduced into these social platforms to provide useful information about the perceived quality or trustworthi-

ness of the textual content/users. Additionally, several moral features of social networks' data can provide new perspectives to tackle the challenges in the text mining community/social networks' data analysis, e.g., hate speech and offensive language differentiation.

1.3.1 Boosting Features

To address the challenges mentioned in section 1.2, several solutions have been proposed. Recently, link analyses and information in microblogging services are used for event detection [70]. Using users' metadata information as new features has also been successful in text processing and unlocking communities [100]. In addition, using Part of Speech (PoS) tags has helped enhance the NLP techniques to comprehend human language [92]. Furthermore, semantic background knowledge can provide conceptual representations and thus improve cluster purity in text document clustering [53].

Another set of social and human-based features can also improve text mining procedures on social networks. For instance, emotion recognition based on known psychological standpoints, as well as the social dimensions of emotion such as emotion transitions and emotion patterns in conversations have been helpful [60]. The sentiment tokens associated to social media posts have also been useful in several studies [7, 77]. Finally, the ideological and stance-based features have been helpful to extract information from social network debates and question-answers [94].

1.3.2 Moral Features in Social Networks

Among several factors that may affect one's social behavior, moral inclinations, which were the first insights in intellectual history [45], play an important role in one's attitude and social interactions with others. According to the sociologist Christian Smith "humans are moral, believing, narrating animals" [45, 93]. However, there is little work studying from the moral aspect of the online reviews.

There have been several studies that model social phenomena based on the insight provided by the moral values extracted from the user-generated content. [105] modeled ideological tendencies

by highlighting morally sensitive issues such as same-sex marriage. In [37], moral sentiments were obtained using word embedding methods and the moral rhetoric over time was extracted to examine the evolution of the moral tendencies. [58] studied morally sensitive datasets and the moral loadings of vice keywords in daily tweets. Some studies employed this technique to associate people's social distances with their moral loadings [29].

We believe investigating the relationships between a user's moral inclinations and her behavior in social networks can help understanding the moral, psychological, and cultural intricacies of human nature that potentially affect their online activities. In this way, we can simplify some of the complications of human interactions by analyzing the moral concerns involved in these interactions.

1.4 Overview

The high-level overview of our structured moral approach combined with social networks' text mining and analyses will be discussed in this section.

1.4.1 Users' Rating Behavior in Online Review Systems

None of the previous studies on the analysis and quantification of morality in social networks have quantified moral loadings for online reviews, nor the way they influence people's rating behavior when any immoral practice is involved.

In this work, we address this issue by studying reviews on Yelp.com, which is a popular online social platform for rating businesses, to investigate people's rating patterns in online reviews as well as how individuals' moral inclinations affect their ratings. This study will reveal the importance of a moral perspective in the analysis of social networks due to concrete moral behaviors on these social platforms. In particular, we are interested in three research problems:

1. If the reviewers' ratings change in the face of moral violations and how this change manifests itself in each moral foundation.

2. If morally-inclined reviewers tend to elicit the same tendencies in their general average rating.

3. If the moral loading of the users is an important factor to study their average rating behavior.

To answer these questions, we apply Doc2Vec, a Natural Language Processing (NLP) technique and Moral Foundations Theory (MFT) [46, 47], a leading conceptual framework in moral psychology. As defined by MFT, a given text can be moral if it contains one or more moral values or non-moral. Using Doc2Vec, we analyze the semantic and syntactical meaning of textual content, and identify reviews with morality (or immorality) associated content. For moral-related reviews, we associate it with each of the five moral foundations to calculate its moral loadings. In this way, we can understand the moral concerns expressed in a review and quantify the moral inclinations of the reviewer.

1.4.2 Social Networks' Abusive Content

Social networks' popularity means millions of people are socializing over these social platforms. Consequently, there is an increase in the negative implications of these online social interactions such as exploiting social media to spread degrading and abusive language. Hate speech and offensive language are two of the most prominent examples of these negative consequences.

Several countries have already taken preventive actions against hate speech, such as United Kingdom, France, and Canada [25] as it poses several ethical and social issues [89].

There have been studies to detect hate speech on social networks due to its social, ethical, and potential legal consequences. For instance, supervised methods are used in [18, 102]. Some studies are based on bag-of-words approaches and discuss this scheme's disadvantages in presence of specific slurs [18, 63]. Another set of studies focus on the syntactic and grammatical features of hate textual data [39, 92, 103]. Finally, there are studies that suggest using a set of different boosting features such as web hyper-links and users' meta data to detect hate communities on social networks [21, 25, 97, 102].

However, most of these studies conflate hate speech and offensive language while they are of different contexts, purposes, and consequences. In addition, none of these studies have focused on the ethical aspect of hate speech and offensive language to better understand the ethical issues they might pose.

In this work, we address these issues by studying a set of labeled tweets based on moral foundations. We first define the differences between hate speech and offensive language and then answer several research questions. In particular we are interested in four research problems:

1. To understand the importance of moral loadings on social networks based on statistical metrics. We examine moral loadings' break down for hate speech and offensive language across several moral foundations, and if these moral foundations are correlated. In particular, we are interested in these correlations' binding and individual aspects.

2. If there is a similar correlation pattern in two different datasets of completely different contexts.

3. If we can use the quantified moral weights as new boosting features to improve the differentiation, classification, and prediction of social networks' textual data, i.e., we use hateful and offensive tweets to test this hypothesis. We examine if we can use the moral features of the tweets as boosting features to improve a baseline approach of mainly conventional tf-idf features.

4. The reasons behind the performance of the improved model which is the baseline approach combined with the moral features, as well as a comparison between this model and the models that incorporate the embedding document vectors.

We employ the Doc2Vec embedding tool to analyze the semantic and syntactical meanings of textual content, and identify reviews with morality (or immorality) associated content. We calculate the moral loading of each tweet with the vice moral words in each moral foundation represented in a dictionary corresponding to MFT. We then use a cosine similarity measure to

identify tweets in each moral foundation. Finally, we classify the tweets by comparing various models with (or without) the moral features. Our results show that moral loadings are helpful in terms of analysis and differentiation of hate speech and offensive language.

1.5 Contributions

The main contributions of this proposal are planned to be:

1. **Utilize the knowledge extracted from text and moral standpoints to analyze social networks, in particular, users' rating behavior.** We first propose a structured method to represent social networks' user-generated content as vectors using NLP embedding tools. We then use the moral aspects of this content to identify the moral reviews and moral break down of the content to understand the rating behavior of the users in the face of moral violations. We then compare moral-concerned users to regular users and study their general average rating.

2. **Utilize the moral attributes as boosting features to classify abusive user-generated content.** We first apply the embedding approach to represent social posts in vectors. We then integrate the moral theory with the vectors to identify the moral break down and moral loadings of the data. Finally, we use the computed moral features to improve the classification of hateful and offensive tweets.

3. **Utilize the moral loadings to understand the moral correlations in two different social contexts.** We apply a moral theory to two different textual datasets: The Yelp reviews and the tweets to understand the moral correlations in two different contexts.

1.6 Thesis Organization

This thesis is organized into the following chapters:

- Chapter 1: Introduction - An introduction to the problem and the main differences between our approach and previous studies.

- Chapter 2: Background - Introducing the main text mining concepts used in the following chapters.
- Chapter 3: Modeling Morality - An overview of our moral theory.
- Chapter 4: Dataset - Yelp dataset, hateful and offensive tweets' statistics and preprocessing.
- Chapter 5: The Proposed Approach - Detailed description of the proposed framework for incorporating morality in NLP.
- Chapter 6: Analysis based on Moral Foundations - Detailed description of analysis of Yelp users' rating behavior based on moral foundations and Doc2Vec embedding method.
- Chapter 7: Hate Speech and Offensive Language based on Moral Foundations - Detailed description of our method to examine moral features as boosting features to predict social media data. We analyze and classify hateful and offensive tweets utilizing the moral perspective.
- Chapter 8: Conclusions and Future Work - The conclusions drawn from the previous chapters and the potential future work.

Chapter 2

Background

Public social networks are indispensable resources of various types of data and interactions. Figure 2.1, introduces users' activities in social networks. These activities result in heterogeneous social networks' data. Richthammer et al. [84] divide the data available on social networks into several categories. In particular, login data, mandatory user profile data, extended user profile data, network data, ratings and interests such as numbers of up-votes, down-votes, votes for coolness, funniness, helpfulness, and users' online business ratings. In addition, the private communication data/disclosed data category might be user inputs such as posts, messages, tagging, following, re-tweeting, comments, and online reviews' information while incidental/disseminated data corresponds to other users' data. We can also observe application data corresponding to behavioral data, and connection data. There are several methods to extract knowledge from the abundant and heterogeneous data on social networks, such as news analysis, scraping, data visualization, opinion mining, sentiment analysis, link analysis, graph-theory based approaches to understand the social graph/the friendships formed based on users' communications, and clustering algorithms for unlocking social media communities. Moreover, there are several supervised/unsupervised/semi-supervised methods available to predict user behavior on social networks.

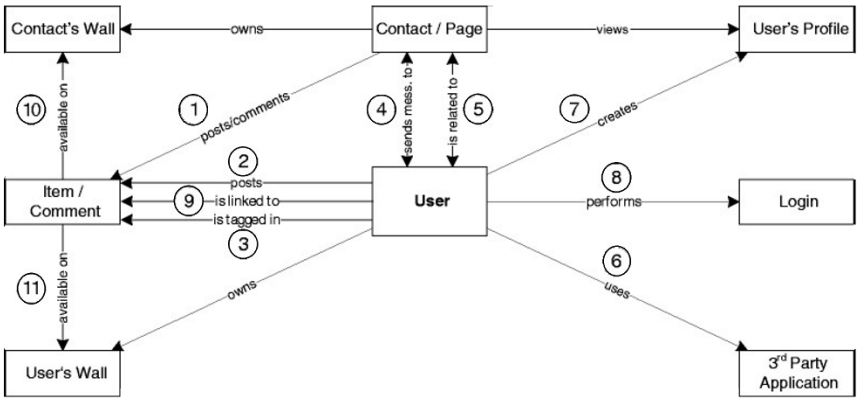

Figure 2.1: User activities on social networks [84]

2.1 Text Mining Methods

Textual mining methods are one important approach among various strategies and algorithms to extract knowledge from the social data. Social networks' interactions are mostly based on human language; however, computers comprehend very little of the meaning of human conversations. In order to be able to command computers to do different tasks and enable them to analyze and understand human language/text to report their results, we have to represent human text in an understandable language for computers. Vector Space Models (VSM) and Distributed Vector Models are two semantic technologies to tackle this problem [98].

2.1.1 Vector Space Models

In vector space models, we compute a word/term-document matrix where each row represents a word in the vocabulary and each column represents a document from a collection. This model is based on the occurrences of a word/term in the document. This scheme is a count vector where the ordering matters since in this model, the first dimension refers to the occurrences of a specific word in all documents. The word–context matrix is term–document matrix as a special case where the word is a chunk of textual data instead of a word. Vector space models also consider pair–pattern matrices where the rows correspond to pairs of terms and the columns represent the patterns where

the pairs occur [98].

2.1.1.1 Tf-idf

The traditional tf-idf, a well-known vector space weighting scheme, which is the short form of term frequency-inverse document frequency is popular in the text mining community. Tf-idf is based on the term frequency count for a specific word or term in each document in the entire corpus. This count is then compared to the inverse document frequency count after normalization. The inverse document frequency count is the count of word (or term) in the whole document [71, 91], i.e., given word w in document d, and a collection of documents D:

$$TF - IDF(w,\ d) = f_{w,\ d} \times log(\frac{N}{f_{w,\ D}}) \tag{2.1}$$

where N is the number of all documents, $f_{w,\ d}$ is the number of the times w appears in single document d, and $f_{w,\ D}$ is the number of documents with the word w [90].

The similarity between two document vectors can be the relationship between the vectors. However, this relationship does not have any semantic or syntactic attributes and is solely based on the occurrences of the word/terms [72].

Tf-idf features will be used in chapter 7 for evaluating various models for the hate speech and offensive language classification task.

2.1.1.2 Bag-of-words

In this model, each document is treated as a bag which contains all words/tokens that appear in the document, disregarding grammar and order. I.e., in this model, word order is not important and we represent documents as unordered lists of words/terms [56]. BoW constructs a matrix where each row corresponds to a word, each column represents a document and each cell is a word count which results in a very sparse matrix. In the BoW approach, "A likes B." and "B likes A." are represented similarly as ["A", "B", "likes"].

2.1.1.3 Point-wise Mutual Information

Point-wise mutual information (PMI) is an alternative weighting scheme to tf-idf where PMI is a measure of association between a target word and a specific context word. Alternatively, PMI is a measure to compute the probability of two words to occur together compared to the situation in which they were independent. This can be applied to an output word w and a context word c as:

$$PMI(w,\ c) = log_2 \frac{P(w,\ c)}{P(w) \times P(c)} \tag{2.2}$$

where $P(w,\ c)$ shows how often we observe these two words occur together, and $P(w) \times P(c)$ indicates how often we expect these two words to co-occur if they were to occur independently. This approach results in a measure that indicates the probability of these words to co-occur compared to the chance probability. PMI's range can be from positive to negative infinity. More positive values imply a higher probability of co-occurring compared to the chance probability while negative values imply less chances of co-occurring [22, 23, 56].

2.1.2 Distributed Vector Models

Distributed vector models are more advanced than the space vector models since the meaning of a word is extracted based on the context words surrounding the word, i.e., similar positions of words in different documents covey similar semantic or syntactic meanings. The distributional hypothesis behind these vector models refers to the idea that words occurring in the similar contexts tend to imply similar meanings [49, 75].

So instead of one-to-one relationships between elements in the vectors and words/terms, each word is represented across all vector elements and contributes to other words' meanings.

2.1.2.1 Doc2Vec

Word2Vec, a distributed vector model, is a word-embedding method of natural language processing recently developed by Google [67]. It is a two-layer neural network to vectorize the words based

on the given text context. Word2Vec performs a skip-gram and bag-of-words approach to do the word embeddings. It returns the words and their corresponding vectors in the semantic space, in which similar words are closer to each other [40, 74, 75].

Doc2Vec is an extension to Word2Vec, which improves Word2Vec by enabling representation of paragraphs and longer blocks of text as individual vectors. Besides the word vectors, a new paragraph vector is defined for every paragraph. Similar to Word2Vec, Doc2Vec is the continuous distributed vector of representations for pieces of texts [67].

In Word2Vec, the aim is to predict a word given its surrounding words. Given a neural network of only one hidden layer, the input IDs are the context words which are the words surrounding the output word. The output layer is the word of interest for prediction. The neural network tries to learn and adjust the corresponding weights by performing the training process to maximize the probability of the output word. These weights will be the vectorized representation of the words after several rounds of training. Doc2Vec follows the same pattern; however, it has additional nodes as special tokens to symbolize each document. Figure 2.2 shows this process where we have an ID for the paragraph and the context words 'the', 'cat', and 'sat' are the input words [67]. If we represent the feature that symbolizes the document contexts as D, the context words as W which are the words in a window surrounding the output word, and the output word as O, Doc2Vec's goal is to maximize the following log probability:

$$max(\sum_{\forall(O,\, W,\, D)} logP(O|\, W,\, D)) \tag{2.3}$$

This stage provides us with the document embeddings and the word embeddings of the training corpus. The second stage is "the inference stage" for the documents that we have not seen yet. This process is similar to the previous maximization step. However, in this stage we can keep the weights as constants and then learn D for the testing corpus [67].

Similar to Word2Vec, Doc2Vec has two versions – the distributed bag-of-words paragraph vectors model (i.e., PV-DBOW) model and the distributed memory paragraph vectors model (i.e., PV-DM). PV-DBOW is similar to the Skip-gram model in word vectors, however, it replaces the

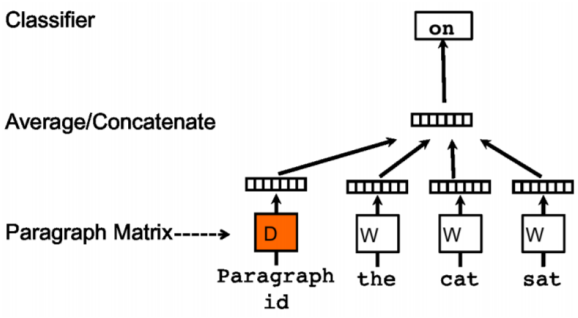

Figure 2.2: Structure for learning document vectors, “the,” “cat,” and “sat” are the context words while “on” is the output word. In this model, the concatenation or average of the current vector and three context words is used to predict the output word [67].

Tf-idf	$f_{w,\,d} \times log(\frac{N}{f_{w,\,D}})$
LSA/LSI	$SVD(f_{w,\,d} \times log(\frac{N}{f_{w,\,D}}))$
LDA	LSI + Dirichlet Prior

Figure 2.3: Comparison among tf-idf, LSA, and LDA

input by a specific paragraph token that symbolizes the documents. Unlike PV-DBOW that ignores the order of the words, PV-DM takes the word order in a small context into account so that important information of a paragraph is preserved. The paragraph token acts as a memory of the context, which is sampled from a sliding window over the paragraph. The paragraph vector can be constructed as either the concatenation or the average of the words in the context, known as the Distributed Memory Paragraph Vector model with concatenated (DMC) or averaged (DMM) paragraph vectors, respectively.

It has been shown that the PV-DM model performs better than the PV-DBOW model because the latter ignores the context words by directly using random initialized words sampled from paragraphs [67]. Therefore, we adopt the PV-DM model as the word-embedding method in this work.

2.1.2.2 Additional Embedding Approaches

There are several other options for the embedding textual data since natural language processing embedding tools have a rich history of studies. Doc2Vec by Le and Mikolov [67] and Word2Vec by Mikolov et al. [75] are two of many embedding techniques.

- Deerwester et al. studied latent semantic analysis (LSA)/Latent semantic indexing (LSI) which is built on top of a VSM approach but adopts the distributional vector hypothesis. LSA assumes that words that have similar meanings will occur in similar blocks of text. LSA uses tf-idf as the weighting scheme and models the documents by representing the corpus in a dimensionality-reduced context matrix. The dimensionality reduction phase is performed by a truncated singular value decomposition (SVD) [28, 64]. SVD converts the high-dimensional sparse word-context matrices into low-dimensional matrices by preserving the semantic relationships [56].

- Latent Dirichlet Allocation proposed by Blei, Ng, and Jordan, which is a three-level hierarchical Bayesian model for modeling items on top of a set of topics, is another option in document modeling [13]. This approach is based on a top modeling perspective and is similar to LSA/LSI, however in LDA, each document is viewed as a combination of various topics. Therefore, a set of topics are assigned to each document by LDA. Moreover, LDA assumes the topic distribution is a sparse Dirichlet prior. The Dirichlet priors assume each document has a limited set of topics which utilizes a limited number of frequent words. An overview and comparison among tf-idf, LDA, and LSA methods is shown in Figure 2.3.

- Another study of Pennington, Socher, and Manning proposed "GloVe: Global Vectors for Word Representation", is based on a word-embedding method that utilizes dimensionality reduction on the co-occurrence word-context matrix [81].

2.2 Summary

In this chapter, we first introduced various existing data and methodologies for social networks analysis. We introduced VSMs and distributed vector models as two semantic technologies for mining natural language in social networks. We then explained the idea and mathematical equations behind VSMs, in particular, tf-idf. Next, we stated the details of the distributional hypothesis and the difference between VSM and distributional approaches in representing textual data as vectors. Finally, we discussed other existing embedding methods such as LDA, LSA/LSI, and GloVe.

Chapter 3

Modeling Morality

3.1 Introduction

Morality coexists within cultural values and psychological inclinations. With a "binding" approach towards morality, moral systems bind people together by concepts such as family, group, and nation, and *ingroup* in general. For example, some moral systems value groups above individuals and consider suppressing individual desires as virtues such as *purity* and *authority*. Other moral systems disregard groups but emphasize the "individuals" welfare by employing moral foundations such as *harm* and *fairness* [45]. Purity is also the main basis for religious laws and the main morality virtue to distinguish moral boundaries [85]. Such moral values represent people's emotions. If a person is inclined to a specific moral virtue, they will feel glad if that moral foundation is practiced or supported [44]. Otherwise, they will feel anger and contempt if a moral virtue is disregarded [86].

Moral inclinations have been studied to distinguish political parties based on the concepts each party tends to endorse. For example, while liberals tend to endorse harm and fairness, conservatives believe in all moral virtues with less emphasis on harm and fairness [43, 46]. Similarly, moral values and individuals' moral inclinations influence their expression of opinions, for example, in terms of ratings in online reviews or the moral weight of the abusive online content in social

networks.

3.2 Moral Foundations Theory

Individuals hold their own moral values to determine right and wrong, however, the definition of moral or immoral vary widely due to contextual and cultural differences. To understand why morality varies across cultures and extract the similarities, MFT explains morality varies as a function of five moral factors, namely moral foundations (MF):

1. Harm (care) as disliking others' pain
2. Fairness as doing justice based on common rules
3. Ingroup as being loyal to one's family or nation
4. Authority as respecting and obeying rules and traditions
5. Purity as feeling aversion towards repulsive things [58].

In MFT, a dictionary consisting of keywords and their stems related to the five moral categories, known as the Moral Foundations Dictionary (MFD) [2], is proposed by Haidt and Graham to represent each moral foundation with a set of keywords [43, 47]. MFD divides each category into vice and virtue keywords. Virtue keywords support their corresponding moral foundation, e.g. "shelter" or "protect" for the harm virtue. Similarly, vice keywords incorporate the words that violate the moral virtue, e.g. "suffer" or "hurt" for the harm vice. In this work, we use 149 vice keywords in MFD.

3.3 MFT in Social Networks

Moral foundations' effects on social networks has been studied extensively in the previous studies. Dehghani et al. [29] investigated the influence of purity homophily as a main predictor of social

distances. In addition, Kaur and Sasahara employed big data analysis and word-embeddings to investigate four morally sensitive issues in twitter [58]. In another study, Zhang and Counts used MFT to extract the ideological patterns and predict potential changes with moral foundations as a predicting factor [105]. Landmann and Hess, investigated if specific emotions are elicited by specific moral foundations [65]. Additionally, Garten et al. employed sentiment analysis on a set of tweets and examined the evolution of the moral rhetoric over time [37]. The political positions of people in twitter and their retweeting behavior in inter-community and intra-community retweets was studied based on MFT variables in [88] by Sagi and Dehghani. In [78], moral inclinations' effects on users' rating behavior was studied.

In the series of studies of Sagi and Dehghani, the semantic similarity between keywords of a specific corpus and the MFD keywords are defined as the moral loading for the topic of interest. The moral loadings can then be used as a factor to test various morally-relevant hypotheses [52, 87]. Inspired by MFT, researchers in other communities have used machine learning techniques and NLP to create a user-defined dictionary of words [37, 52]. Furthermore, big data techniques have been used by Boyd et al. to investigate the relationship between moral values and the behavioral patterns in texts [14, 52].

Chapter 4

Dataset

4.1 Yelp Dataset

Yelp.com is a popular online social network for rating businesses. In this work, we use an open dataset from the Yelp Dataset Challenge Round 9 [4], which includes 4,153,151 reviews on various kinds of businesses with each review being rated from 1 to 5 stars.

This dataset also has more than 947,000 tips by one million users for 144 thousand businesses in UK: Edinburgh, Germany: Karlsruhe, Canada: Montreal and Waterloo, U.S.: Pittsburgh, Charlotte, Urbana-Champaign, Phoenix, Las Vegas, Madison, and Cleveland. The dataset is of three main components: the reviews, the businesses, and the users.

The reviews are presented with several features: review ID, user ID, business ID, star rating of the review, date of the review, review text, count of useful votes, count of funny votes, and count of coolness votes.

The businesses are attributed to: business ID, business name, neighborhood name, full address, city, state, postal code, latitude, longitude, star rating, number of reviews for the business, and is open 0/1 (closed/open).

The users have the following features: user ID, first name, review count, yelping since, IDs of friends, number of useful/cool/funny votes sent by the user, number of fans the user has, an array

of years the user was elite, average stars, number of hot/more compliments/profile compliments/-cute/cool/funny/writer/photos compliments received by the user, and list/note/plain compliments received by the user.

In our study, because we concern about the moral inclinations of the reviewers' text, we first constructed a morality-relevant dataset of 7,039 English reviews by filtering the reviews with keywords "moral" and "ethic" by using the review text.

In our data preprocessing, we removed all extra white spaces and punctuations in the text and converted the capital letters to small letters to avoid extra preprocessing for the uppercase letters.

We then used the reviews' star ratings and user's average star rating as features in our studies in chapter 6.

4.2 Hateful and Offensive Tweets

We use the labeled data provided by Davidson et al. [25], which contains 24,783 tweets. The tweets were gathered from the twitter API by searching specific hate lexicons from *Hatebase.org* and extracting the timeline for the users resulting in 85.4 million tweets. A sample of these tweets was then labeled by CrowdFlower [1] workers as 'hate speech', 'offensive language', and 'neither' after receiving instructions on correct strategies for annotation of the tweets, i.e., the workers labeled the data based on the context in which the words were used and not necessarily by detecting a specific slur. Each tweet was labeled by at least three workers and the inter-agreement of 92% was reported by CrowdFlower. Each tweet was assigned a final label of hate speech, offensive language, or neither based on a simple majority vote [25]. The number of tweets in each label is listed in Table 4.1.

In the preprocessing step, we lowercased and stemmed the tweets using the Porter stemmer. We removed extra white spaces and the punctuation.

Table 4.1: Number of tweets in each category

Category	Count of Tweets
Hate Speech	1,430
Offensive Language	19,190
Neither	4,163

4.3 Yelp Dataset and Tweets' Discrepancy

Both the Yelp reviews dataset and the tweets' dataset contain a collection of short texts. But they differ from each other in the following aspects:

First, these two datasets are from different contexts: Yelp.com is known for rating various businesses such as restaurants, therefore, our dataset contains online reviews discussing the pros/cons of each business. While our tweets are related to hate speech and offensive language and mostly contain curse words.

Second, the average length of the textual data is different in both datasets. Twitter has a 140-character limitation policy restricting the length of its posts which results in very short texts. Therefore, the average word count of our tweets is 13.63 words; however, Yelp does not have this limitation and users can discuss the businesses in details. Therefore, the average word count of our Yelp dataset is 269.097.

Considering different characteristics of the tweets and the Yelp reviews, we can conclude that we are studying two different sets of social networks' textual data.

Chapter 5

The Proposed Solution's Framework

5.1 The System Design Rationale

We build our system based on Doc2Vec for the analysis of the Yelp dataset, and tf-idf and Doc2Vec for tweets' analysis and prediction. There are several reasons behind choosing Doc2Vec from several approaches discussed in chapter 2 such as LDA, LSA, Word2Vec. Doc2Vec captures the semantics of the document based on a full block of text in the entire corpus. Analysis based on words which is the main concern is the previous embedding approaches, cannot efficiently capture the semantic similarities of the documents as a block of text relative to other blocks of text in the corpus. In addition, several studies have shown Doc2Vec's superiority. For instance, Campr and Ježek [19] compared Doc2Vec, Word2Vec, LDA, and LSA. The results demonstrated Doc2Vec has superior performance. In [66], it was demonstrated that Doc2Vec performs better than Word2Vec by comparing these approaches performance and adopting several short and long corpora. The original paper on Doc2Vec by Le and Mikolov [67] also showed its superiority compared to other embedding methods, such as a specific version of LDA.

Tf-idf is a traditional embedding approach in text mining community. It is popular due to several advantages such as easy computation, inclusion of the most descriptive words based on a simple weighting scheme, finding word overlaps for longer documents of nearly thirty words [26].

Therefore, it is our main choice for providing a set of fundamental features and a baseline approach which will be used for further enhancement by incorporating additional features.

5.2 Semantic Similarity to Moral Foundations

The Doc2Vec model learns paragraph vectors from unlabeled text data of a variable length, which makes it an attractive method to process the textual content of online reviews in chapter 6 and the content of tweets in chapter 7 in this work. Therefore, we adopted the PV-DM model to convert each online review (tweet) as a document to a vector in the semantic space. In particular, we implemented the DMM approach and utilized the average word vectors of the key words of each moral foundation. Meanwhile, words in the reviews (tweets) are represented as vectors in a vector space where semantically similar words have similar vector representations. In this way, we can calculate the text similarity of the review (tweet) to a moral foundation as the cosine similarity of the document vectors and MF words by averaging MF words' respective vectors.

5.3 The System Architecture

In this section, we will provide a step by step description of our proposed scheme. The flowchart of this scheme is shown in Figure 5.1.

5.3.1 Data Preprocessing

The performance of a pattern recognition system is bound to an appropriate data-preprocessing technique. For the Yelp dataset preprocessing, we removed all extra white spaces and punctuations in the text and converted the capital letters to small letters to avoid extra preprocessing for the uppercase letters.

Similarly, for the dataset of tweets, we first lowercased the tweets. We then stemmed the tweets using the Porter stemmer. Moreover, we removed extra white spaces and the punctuation. We also

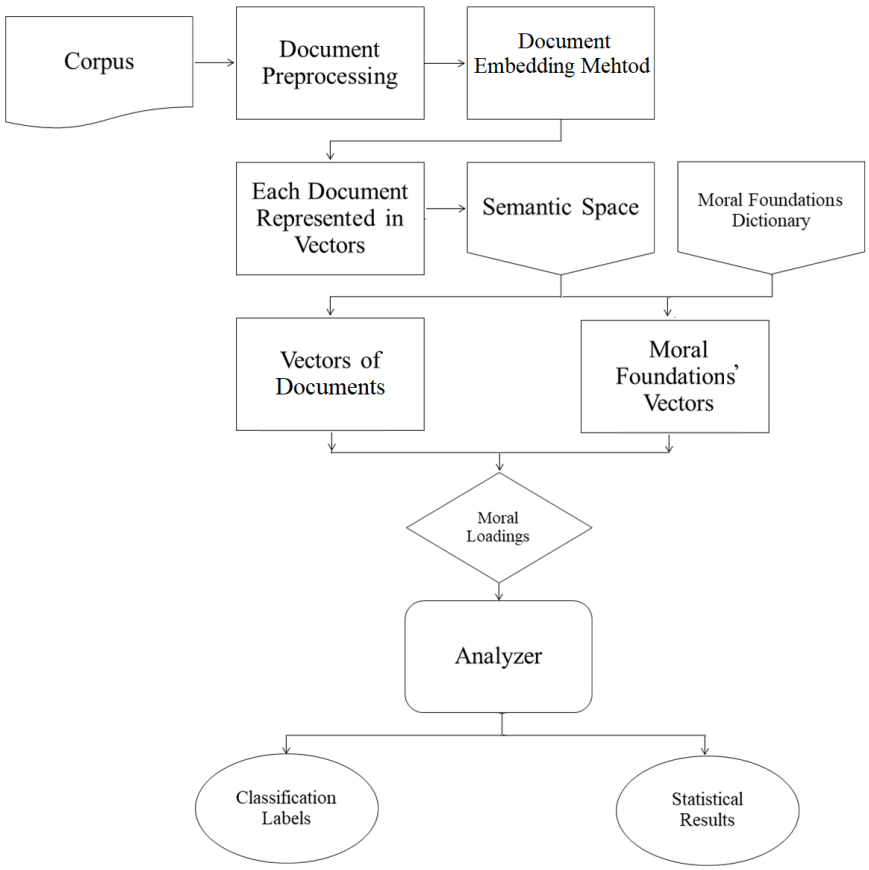

Figure 5.1: Structure for learning document vectors and computing moral loadings

removed twitter's special tokens: @, RT, hash signs, and URLs only during Doc2Vec model's training step.

5.3.2 Compute Document Vectors

We first tokenized the corpus based on the whitespace and removed non-alphanumeric characters and the less frequently occurred tokens. Then, we treated each document as a separate paragraph to train the PV-DM model.

After training, we obtained the vector representations for each token in the corpus. For each sentence, we formed a vector, $r = (r_1, \ldots, r_v)^T$, corresponding to the summation of the vectors of all the tokens in the sentence. Similarly, for each of the five moral foundations, we formed a vector, $f = (f_1, \ldots, f_v)^T$, corresponding to the summation of the vectors of all the vice keywords

in that moral foundation. These vectors are the representations of our documents/moral words in the semantic space.

Some studies have proposed the potential to represent the words/documents in Euclidean space instead of a semantic space [50].

5.3.3 Compute Moral Loadings

We used the cosine similarity between a document vector r and a moral foundation vector f as a measure for the similarity of a document to a moral foundation, which is calculated as the dot product between the two vectors normalized by their norms. In the moral context, a cosine similarity close to 1 indicates that the review is semantically similar to that moral foundation, while a cosine similarity close to 0 indicates that the document and the moral foundation are not semantically related.

Definition of Moral Loading - We define the moral loading m_{ij} for a document as the cosine similarity between the moral foundation f_i the document r_j, where $i \in \{1,2,3,4,5\}$ is an index representing each moral foundation and j corresponds to a document's index in the entire corpus. The quantified moral loadings were inspired by a series of studies of Sagi and Dehghani [87, 52] and Dehghani et al. [29] where the semantic similarity between a corpus's keywords and the MFD keywords are defined as the moral loading for the topic of interest.

We then use the moral loadings as features to distinguish documents in each moral foundation and feed them to an analyzer to either do a classification task or statistical analyses. To quantify the moral loadings using the Doc2Vec method, we adopted Python's genism module [83] to learn paragraph vectors.

Finally, we used cosine similarity to quantify moral loadings. However, there are other similarity measures which can be utilized to do this task, such as:

- Block distance or Manhattan distance, compares the distance in a grid path to get from one point to another data point. The Block distance between two data points is the sum of the

differences of their corresponding components [41, 61].

- Dice's coefficient is computed as twice the number of common terms in two documents over the total number of terms in both documents [30, 41].

- Euclidean distance or L2 distance is the square root of the sum of squared differences between corresponding elements of the two vectors [41].

- Jaccard similarity is defined as the number of common terms divided by the number of the unique terms in the documents [41, 55]

- Matching coefficient is a vector-based scheme where we count the number of similar terms in the documents where both document vectors are non-zero [41].

- Overlap coefficient is very similar to Dice's coefficient; however, if one document is a subset of the other document, we will consider the similarity as a full match [41].

In addition, there are character-based similarity approaches such as Longest Common SubString (LCS) algorithm which considers the similarity between two strings as the length of contiguous chain of characters that are common in both strings, or N-grams where the similarity is defined as the count of the common N-grams in two strings over the maximal number of the N-grams in two strings [9, 41].

5.3.4 Analyzer

In this section, we introduce the measures we used to understand the importance of a moral loadings in social networks' data based on a statistical analysis approach. We employed two measures: Conditional Relative Frequency and the correlations between the moral foundations.
Lastly, we will discuss the classification algorithm options.

5.3.4.1 Conditional Relative Frequency

If the dataset we are trying to analyze is imbalanced, a simple frequency analysis cannot efficiently represent the true statistics of the dataset. For instance, when we studied the original dataset of 4,153,151 Yelp reviews, we found that the dataset has an unbalanced number of ratings. The number of reviews rated with 5 stars is much larger than the reviews rated with 1 star. In particular, there are 1,704,200 reviews or 41% of the entire Yelp reviews with a 5-star rating and 540,377 reviews which is equal to 13% of of Yelp reviews with a 1-star rating.

Due to this imbalance, we believe the conditional distribution should be more informative and reasonable than the direct distribution. Therefore, if we represent the star rating in each moral foundation as r and the dataset of interest as D, we defined the conditional relative frequency (CRF) of each rating as:

$$CRF(r,\ MF_i) = \frac{f_{r,\ MF_i}}{f_{r,\ D}} \quad s.t. \quad i \in \{1,2,3,4,5\} \ \ and \ \ r \in \{1,2,3,4,5\} \tag{5.1}$$

where D is the dataset of interest, i is an index for each MF, r is 1 to 5 star ratings, $f_{r,\ MF_i}$ is the number of star ratings in each MF, $f_{r,\ D}$ is the number of star ratings in the dataset of interest, and $CRF(r,\ MF_i)$ is the conditional relative frequency of rating r for moral foundation i.

- **Yelp reviews' dataset of interest** can be the entire Yelp dataset or the morally filtered dataset of 7,039 reviews.
- **Offensive and hateful tweets' dataset of interest** is the entire corpus of 24,783 tweets since the dataset is not filtered with specific keywords.

For example, in the case of Yelp reviews, if we have 264 reviews with a 5-star rating in the harm MF and 2,211 5-star reviews among 7,039 moral Yelp reviews, the CRF will be defined as:

$$CRF(5,\ MF_{harm}) = \frac{264}{2211} \times 100$$

which means there are more than 11% reviews in the 5-star rating considered related to harm. In chapters 6 and 7, we will discuss the process of identifying reviews/tweets in each MF.

5.3.4.2 Our Correlation Measure

To understand the moral correlations in our datasets we studied the correlations between the moral foundations. These correlations will help us extract hidden moral relationships in our data and to examine if specific MFs are correlated based on their binding or individual attributes. We can also compare the moral correlations for two different textual and contextual datasets (Yelp reviews and hateful/offensive tweets).

There are many correlation measures that can be used to calculate the moral correlations:

- Spearman's rho or Spearman's rank correlation coefficient which is a rank-based monotonic function. A high correlation conveys similar rankings between observations within two variables [68].

- Kendall's Tau or the Kendall rank correlation coefficient, is a statistical metric that measures the ordinal association between two variables. This correlation is rank-based, so high correlations imply similar rankings between observations within two variables [59, 62].

- Gamma correlation coefficient is another rank-based correlation where we measure the similarity of the orderings of the data when ranked by the observations' quantities [42].

In this work, we use the Pearson correlation coefficient (PCC) as a measure of correlation. It is also referred to as Pearson's r. Pearson's r is a measure of the linear correlation between two variables and was introduced by Karl Pearson [79]. The range of this correlation is from -1 to 1 where 1 indicates perfectly positive linear correlation, 0 indicates no linear correlation, and -1 indicates perfectly negative correlation. For two variables X and Y, this correlation is computed as below:

$$r_{XY} = \frac{cov(X,\, Y)}{\sigma_X \times \sigma_Y} \tag{5.2}$$

Where $cov(X, Y)$ is the covariance of X and Y, σ_X is the standard deviation of X, and σ_Y is the standard deviation of Y [79].

5.3.4.3 Classification Algorithm

The classification task is to predict tweets' labels based on a moral perspective. We want to examine the moral features applicability as new boosting features to improve the classification performance. Several classification algorithms such as regularization-based algorithms (multi-class/one-vs-all approaches), decision tree-based approaches, Naive Bayes, kNN, etc. can be used to perform this task. In this work, we examined all these algorithms and selected logistic regression with L2 regularization due to its superior performance compared to other models.

5.4 Summary

In this chapter, we described our scheme to understand/analyze moral features in social networks. We first described our rationale for choosing Doc2Vec and tf-idf. We then defined our scheme by introducing our preprocessing strategy, our approach to represent documents in the semantic space using Doc2Vec, and how to calculate the moral loadings. Next, we introduced conditional relative frequency and Pearson's r correlation as two metrics to examine concrete moral relationships in social networks' data. Finally, we discussed several classification strategies to incorporate moral loadings in social media data prediction.

Chapter 6

Rating Behavior based on Moral Foundations: The case of Yelp Reviews

In the previous chapters, we discussed the importance of a moral analysis on user-generated content in social networks and introduced our approach. In this chapter, we will conduct this study by first applying our approach on Yelp reviews. Next, we define a similarity threshold to identify a set of reviews in each moral foundation. We present the frequency and conditional relative frequency of the reviews in each moral foundation to compare the rating behavior of regular users with moral-concerned users. In addition, we examine the correlations between moral foundations as another metric of importance of moral patterns in social networks. Finally, we perform a study for moral-concerned users to compare their moral rating behavior with their non-moral rating behavior reflected in their average rating [78].

6.1 Compute Yelp Reviews' Moral Loadings

The moral loadings are calculated in the following steps:

1. **Represent reviews' vectors in the semantic space**: As described in section 5.3, using Doc2Vec, we converted each review to a vector in the semantic space. We considered each

review as a document and used each review as a separate paragraph to train the Doc2Vec PV-DM model. We embedded the reviews into vectors of size 100 in the semantic space. In addition, we used a window size of 10 and negative sampling of size 5 which indicates the count of the noise words drawn by negative sampling.

After training, we obtained the vectors' representation for each token in the corpus. For each review, we formed a vector, $r = (r_1, \ldots, r_{100})^T$, corresponding to the summation of the vectors of all the tokens in the review. Similarly, for each of the five moral foundations, we formed a vector, $f = (f_1, \ldots, f_{100})^T$, corresponding to the summation of the vectors of all the vice keywords in that moral foundation.

2. **Similarity-based moral loadings**: We used the same scheme described in section 5.3.3. We defined the cosine similarity between a review vector r and a moral foundation vector f as the measure for the similarity of a document to a moral foundation. This similarity is calculated as the dot product between the two vectors normalized by their norms and named moral loading m_{ij} for a review where $i \in \{1,2,3,4,5\}$ is an index representing each moral foundation and j is a review/user's index for identification in the entire corpus.

   Similarly, we define the moral loading M_{ij} for a reviewer u_j as the average of the moral loadings of all his reviews.

6.2 Yelp Users' Rating Behavior

In this work, we conducted two experimental studies to explore the relationship between people's moral concerns and their rating behavior. In particular, we first identified the frequency of the ratings in each moral category and calculated the conditional relative frequency considering the unbalanced datasets. Secondly, we tracked the regular users who have rated the same businesses as the moral-concerned users identified in our moral corpora, and studied the differences in their rating behaviors.

6.2.1 Identify Relevant Users based on the Moral Loadings

In the first study, we aim to investigate if people who care more about morality will rate differently from the regular users who do not show a clear moral inclination in the face of moral violations. We are also interested in exploring the different ways that the reviewers rate under different morality contexts.

To tackle this problem, we first need to identify individuals associated with each of the five moral foundations, i.e., harm, fairness, ingroup, authority, and purity. As described in chapter 5, we calculated the moral loadings in each MF category as the cosine similarity for a review and the keywords in that MF category. We then ranked the reviews based on their moral loadings.

To locate the most similar document to each moral foundation, we defined a cosine similarity threshold. It is pointed out in [82] that the threshold for the cosine similarity measures in document comparison should be dynamically adjusted, since low cosine thresholds can produce good results in terms of precision and recall. In fact, setting a too high threshold without considering the specific context's experimental results will result in excluding documents that are similar. As recommended in [32], "researchers taking the factor analysis approach to LSA should not apply 0.40 or some similarly preset loading threshold, but instead apply an empirically derived threshold, validated by a domain expert because thresholds as low as 0.18 were found acceptable." Following this idea, we experimentally set the threshold for cosine similarity in our moral corpora to 0.2. Our empirical analysis showed this threshold as a good boundary to distinguish morally similar documents. In the morally filtered dataset of 7,039 reviews, there are 5,782 reviews with the cosine similarity larger than 0.2.

In particular, we had 1,002 reviews for the Ingroup MF category, 1,115 reviews for authority, 1,118 reviews for harm, 1,188 for reviews fairness, and 1,359 reviews for purity. In each case, the reviews with a loading above 0.2 maintained a reasonably strong relevance to the respective moral foundation category. In the meantime, the result provides a reasonably large size of morally relevant reviews to be used in further analysis.

6.2.2 Relationship between Users' Moral Concerns and Ratings

Next, we studied the direct relationship between a user's own moral inclinations and her rating behavior. In the Yelp dataset, for each review, a user explicitly gives a star rating, ranging from 1 star to 5 stars. The result is shown in Figure 6.1. For each moral foundation category, we show the distribution of reviews in different star ratings. In all moral foundation categories, it is obvious that the reviews with 1-star rating outnumber the reviews of any other rating. This indicates users who care more about the moral concerns tend to give low (i.e., 1-star) ratings.

However, the original dataset of 4,153,151 reviews, has an unbalanced number of ratings. The number of 5-star rated reviews is much more than 1-star reviews. In particular, there are 1,704,200 reviews with 5 stars which equals to 41% percent of the entire Yelp reviews and 540,377 reviews are 1-star which incorporates only 13% of Yelp reviews. Therefore, we believe the conditional distribution is more useful and reasonable than a direct distribution. We compute the CRF based on equation 5.1.

We first analyzed the conditional relative frequency in relevance to the entire dataset with 4,153,151 reviews. As shown in Figure 6.2, in all five moral foundation categories, the frequency of reviews in 1-star rating is significantly larger than the number of reviews in other star ratings.

Next, we calculated the conditional relative frequency in relevance to our moral dataset (i.e., the dataset with 7,039 moral-relevant reviews). The result is shown in Figure 6.3 For instance, in the moral dataset of 7,039 reviews, there are more than 20% reviews in 1-star rating considered related to fairness, while only 6% reviews in 5-star rating are considered related to fairness.

Moreover, in all moral foundation categories except the purity category, a consistent stepwise decreasing pattern was observed in the conditional relative distribution of reviews with different star ratings. This indicates that users giving lower ratings tend to consider more about the fairness, authority, ingroup, and harm aspects in their reviews, while users giving higher ratings have less considerations in mind [78].

The only exception is the purity category, in which no matter which star rating is given, an

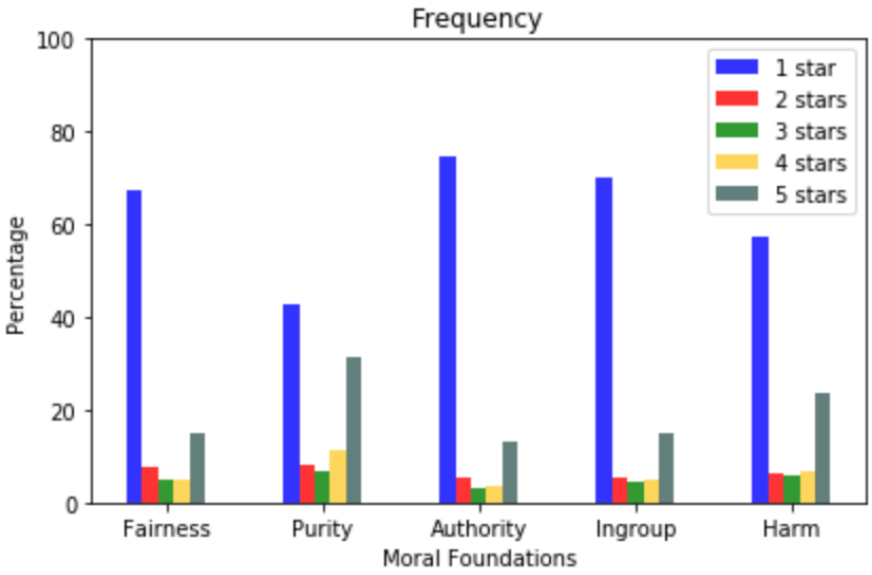

Figure 6.1: Frequency of each rating in five moral corpora

approximately same relative percentage of users care about purity (e.g., "disgust", "gross", "indecent", "trashy", etc.) in the reviews. This finding is in line with some previous work on moral foundations. For example, Dehghani et al. [29] investigated the influence of purity homophily as a predictor of social distances. Their results indicated that comparing with other moral foundations, purity is the main predictor of the social distances.

6.2.3 Moral-concerned and Regular Users' Rating Behavior

In this task, we studied the rating behavior of regular users and users with moral inclinations.

In section 6.2.1, we identified a set of users whose reviews are related to five moral foundations. We also searched the entire Yelp dataset to locate another group of 370,221 users (with repetition), who had reviewed the same set of businesses that the moral-concerned users reviewed. Therefore, we constructed two user sets, e.g., regular users and moral-concerned users.

We first show regular users' rating distribution in Figure 6.4 (left). For the target set of businesses, this figure shows percentages of reviews with different ratings. Overall, there are more reviews with 4 and 5 star ratings than the ones with 1-3 star ratings. We also plot the conditional relative frequency of review ratings in relevance to the count of each rating in the entire dataset,

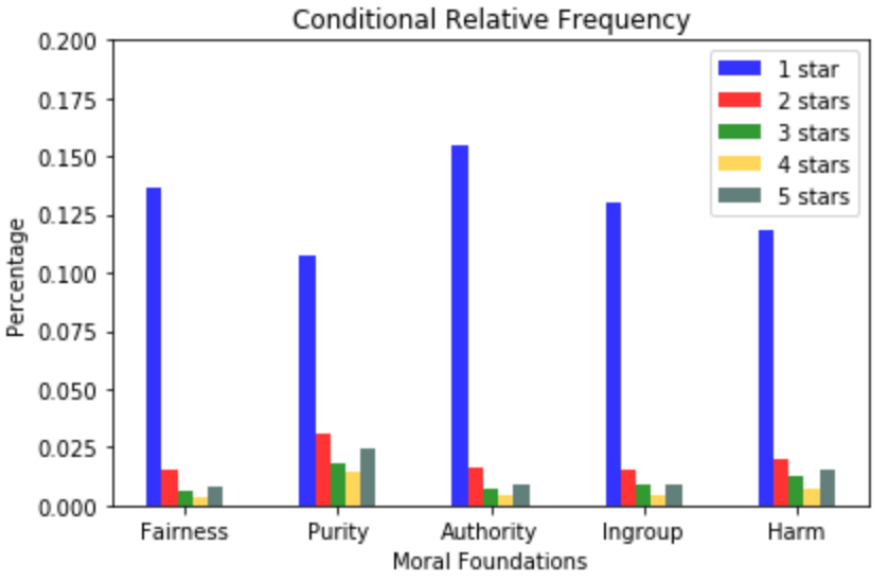

Figure 6.2: Conditional relative frequency of each rating relative to the dataset of 4,153,151 reviews

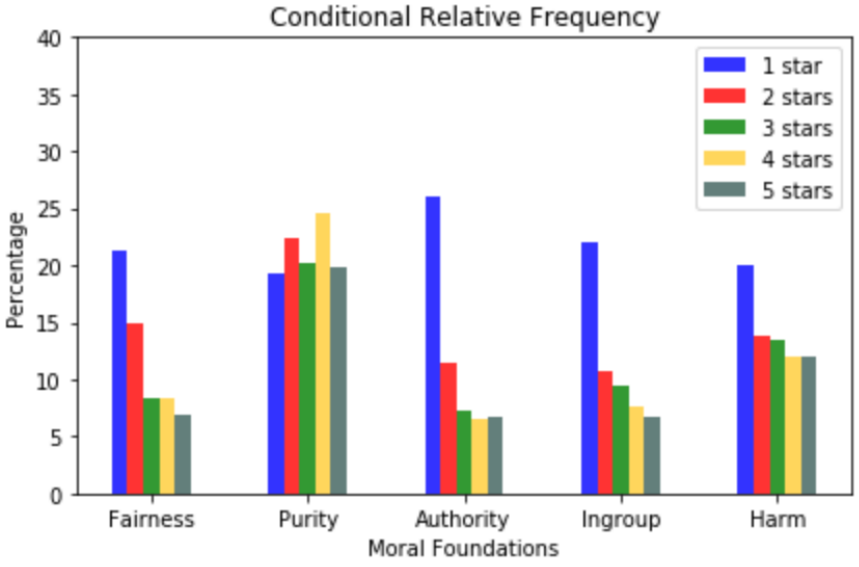

Figure 6.3: Conditional relative frequency of each rating relative to the dataset of 7,039 reviews

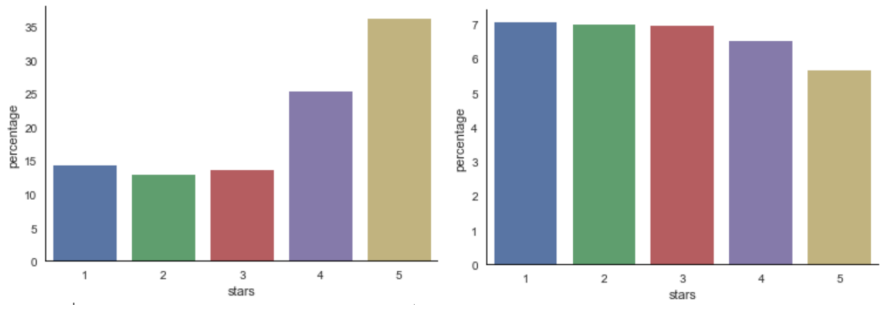

Figure 6.4: Frequency (left) and conditional relative frequency (right) of the ratings of regular users

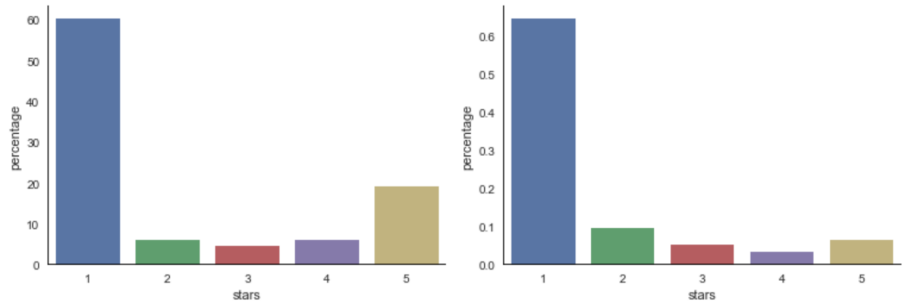

Figure 6.5: Frequency (left) and conditional relative frequency (right) of the ratings of moral-concerned users

as shown in Figure 6.4 (right). This indicates the rating behavior of the selected regular user set is compatible to the general users and the selected regular users are not biased [78].

Next, we studied the rating distribution of the users in the moral set. The frequency and conditional relative frequency of ratings of moral-concerned users are shown in Figure 6.5. In both plots, there are significantly more reviews with 1-star rating than reviews with higher ratings. This is in line with our findings in the previous task.

Comparing Figure 6.4 and Figure 6.5, we clearly see that regarding the same set of businesses,

Table 6.1: Correlations between stars' count and cosine similarities

Moral Foundations	Correlation
Fairness	-0.302
Harm	-0.145
Authority	**-0.350**
Ingroup	-0.284
Purity	0.065

regular users who do not care about the moral foundations or face any moral violations, rate differently from users who do have moral concerns. In particular, people who care about moral violations tend to rate lower than regular users. Moreover, the moral-concerned users tend to give the lowest rating compared to the regular users.

We further studied the correlation between the moral loadings and the count of star ratings for our five moral corpora using Pearson's r which was described in equation 5.2. As shown in Table 6.1, the negative correlations for ingroup, fairness, harm, and authority are compatible with the previous bar plots, since the higher ratings the reviews have, the smaller their moral loadings will be.

It is worth to point out that the correlation result of the purity category also went along with the unbalanced rating pattern of the reviewers in the purity moral corpus [78].

6.3 Moral-concerned Users' Average Ratings and Correlations

In the previous study, we showed that users with moral considerations rate differently from the regular users. In this study, we aim to examine the rating behavior of the moral-concerned users by comparing the average ratings of their moral-related reviews and the other reviews that do not show clear moral relevance. In other words, if the users with high moral loadings show the same moral inclinations in their general average ratings.

6.3.1 Users' Average Rating Behavior

To compare moral-concerned users' average moral ratings and average non-moral ratings, for each user in the moral set identified in section 6.2.1, we calculated the rating difference of the user as:

$$d_{ij} = |r_{ij} - g_j| \quad s.t. \quad 0 \leq d_{ij} < 4 \tag{6.1}$$

where g_j denotes the average rating of all her reviews, and r_{ij} denotes the average rating of her reviews regarding the moral foundation i and j is an index for a reviewer u_j.

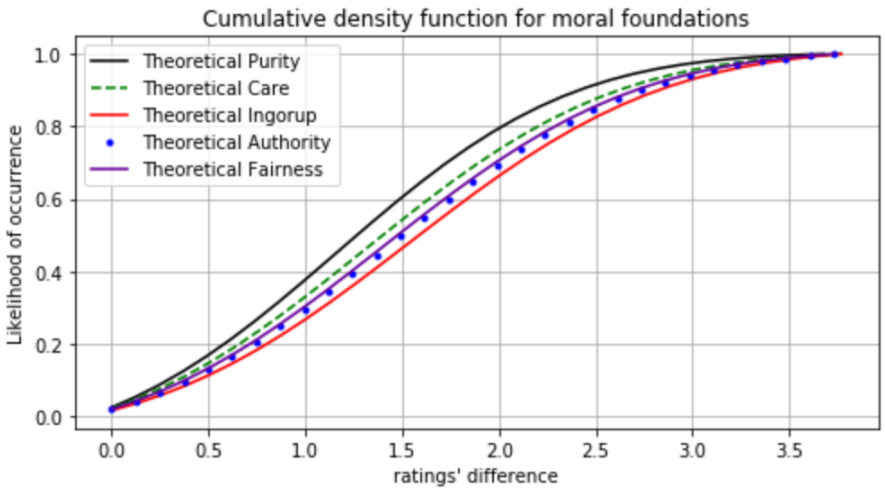

Figure 6.6: CDF of the absolute difference of average moral ratings and general average moral ratings of the reviwers for each moral foundation

For each of the five moral foundations, we calculated the rating difference for all the users with reviews relevant to this moral foundation. The corresponding cumulative density function is shown in Figure 6.6 to provide direct statistical insights about the moral-relevant users' rating behavior. Generally speaking, these reviewers tend to show the same rating behavior in their overall reviews as compared to their moral-concerned reviews. As shown in Figure 6.6, the maximal of the average rating difference is 3.8, and more than 50% of users have a rating difference smaller than 1.5 stars, which is close to the theoretic average rating difference of 2 stars.

Next, we define the weighted average rating difference by incorporating the moral loading of each user as the weight. This is because users related to one moral foundation have different moral loadings, which indicates the degree of inclination to the moral foundations. Consequently, we calculate the weighted average rating difference as $M_{ij} \times |r_{ij} - g_j|$, and show the corresponding cumulative density function results for five moral foundations in Figure 6.7.

As shown in Figure 6.7, over 90% of users have a rating difference smaller than 2.5, and over 20% of users have a rating difference smaller than 1, in all five moral foundations. This indicates that moral-concerned user rates consistently in their moral-concerned reviews and the general reviews [78].

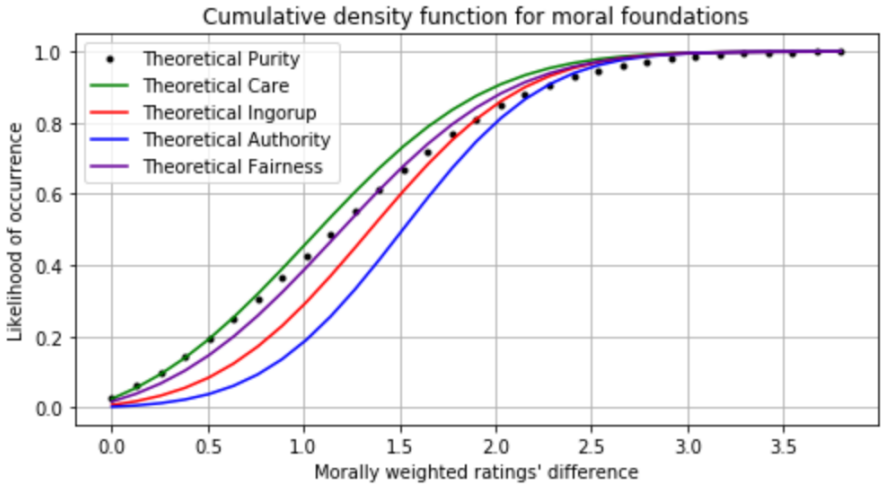

Figure 6.7: CDF of the morally weighted absolute difference of average moral ratings and general average moral ratings of the reviwers for each moral foundation

Table 6.2: Cosine similarities' correlations

Moral Foundations	Fairness	Harm	Authority	Ingroup	Purity
Fairness	-	0.355	**0.681**	0.384	0.286
Harm	-	-	0.368	0.385	0.332
Authority	-	-	-	0.527	0.145
Ingroup	-	-	-	-	**0.132**
Purity	-	-	-	-	-

6.3.2 Correlations between Moral Foundations

We also studied the correlations between moral foundations with Pearson's r defined in equation 5.2. As shown in Table 6.2, authority and fairness have the highest correlation, and purity and ingroup have the lowest value. In fact, purity is not highly correlated with any other moral foundation. This is in line with previous studies [29, 58, 85], our findings in Table 6.1, and the unbalanced ratings in section 6.2, which indicates that purity is the most peculiar moral foundation. We also observed that ingroup and authority are highly correlated. This may be because they are from the binding foundations [45]. We expected a higher correlation between harm and fairness since they are both individualizing foundations; however, this was not observed in our results.

Finally, we compute a word cloud with all vice keywords identified in our moral corpora. If the word is not directly in MFD (e.g., MFD by default has several words for the root 'desert'), we

Figure 6.8: Word cloud of MFD's vice keywords in our moral corpora

present words of the same root under one word of the same root which is available in MFD. E.g., 'favor', 'favors', 'favored' are all of the same root and their representative in MFD is 'favoritism'. As shown in Figure 6.8, keywords 'refuse' and 'favoritism' are highlighted as the most frequent words in our moral corpora.

6.4 Summary

In this chapter, we analyzed the moral loadings of Yelp reviews by employing the moral foundations theory and by using Doc2Vec as an embedding method to represent the documents in the semantic space. We first identified reviews in each moral foundation based on a cosine similarity threshold. Next, we performed a set of statistical analysis using conditional relative frequency and the correlations between MFs to understand the hidden moral patterns in social networks' data, and to compare the rating behavior of moral-concerned users and regular users. Lastly, we examined moral-concerned users' moral and non-moral average ratings.

Our results show concrete moral behavior on social networks. We observed that moral-concerned users tend to rate with the lowest possible rating (1-star) and their rating behavior is different from the non-biased rating pattern of regular users.

Moreover, we observed that purity is the most peculiar moral foundation. Finally, moral-concerned users show the same moral inclinations in their general non-moral average rating.

Chapter 7

Hate Speech and Offensive Language Analysis and Prediction based on Moral Foundations

In the previous chapter, rating behavior of the users was examined by employing moral foundations theory and Doc2Vec. This chapter begins by describing the importance of a similar moral study in terms of user-generated abusive content such as hate speech and offensive language in social networks. We then propose a study to understand and detect hate speech and offensive language based on the moral foundations, which has not been done in previous studies. We first use Doc2Vec and the moral foundations theory to identify tweets in each moral foundation. Next, we discuss the correlations and frequency of the tweets in each moral foundation. Finally, to examine the applicability of moral features in prediction of social media textual data, we use the moral loadings as new features to improve a baseline approach of mainly tf-idf features to classify the tweets.

7.1 Existing Hate Speech and Offensive Language Studies

There is a rich history of studies on detecting hate speech on social networks since hate speech targets victims based on their intrinsic features and can ignite violence. Therefore, some countries have taken legal actions against hate speech [25].

Most studies conflate hate speech and offensive language and refer to it as 'hate speech', e.g.,

there are supervised methods that do not particularly emphasize on the difference between hate speech and offensive language such as [18, 102]. While [25] uses a crowd-sourced hate speech lexicon to collect tweets containing hate speech, as well as a crowd-sourcing platform to label the tweets as hate speech, offensive language, and neither. They employ a multi-class model to automatically classify the tweets and emphasize on the difference between hate speech and offensive language.

Another set of studies have been focused on the applicability of a bag-of-words approaches, e.g., [18, 63], discussed that bag-of-words embedding tools can have a high recall, but at the same time they tend to misclassify hate speech writing due to the presence of specific curse words and slurs.

Some studies looked at the syntactic and grammatical attributes of hate speech textual data. For instance, in [103], grammatical relations were explored for semantically filtering offensive language from a sentence. Several studies have focused on finding the hate groups online. While [39, 92] studied syntactic features and the use of part of speech tags as syntactic features to detect hate speech.

There are studies that focus on different boosting features such the micro-blogging hyper-links and users' meta data to detect hate communities. For instance, in [21], web hyper-links were used to detect hate groups. In [97], Facebook's hate groups' activities were studied based on text mining and social media analysis techniques. While [25, 102] mention that features such as gender and ethnicity in users' meta data are of great importance in detecting hate speech.

[36] uses a deep learning approach utilizing convolutional neural networks for hate classification on social networks.

Finally, as described in [89], hate speech and offensive language are different from cyberbullying on social media. Cyberbullying can have different motives which may not necessarily be related to the intrinsic features of the victim such as their race, gender, sexuality, and ethnicity. Furthermore, cyberbullying is repetitive and is based on a power imbalance between the victim and the bully.

7.1.1 Hate Speech and Offensive Language Detection Methods

There are several methods to detect hate speech/groups and offensive language on social media, a few examples are:

- Bag-of-words and tf-idf approaches
- Syntactic information such as Part of Speech (PoS) tags
- Grammatical features
- Users' meta data
- Supervised methods to classify social networks' textual data
- Web hyper-links and social media edges to unlock hate communities in social networks' structural data
- Deep learning approaches

The details and a summary of these approaches in the existing studies mentioned in the previous section are listed in Table 7.1. The column named "H/O differentiation" with yes and no values, specifies if the paper has differentiated hate speech and offensive language.

7.1.2 Hate Speech and Offensive Language Challenges

Automatic detection of abusive user-generated content on social networks, in particular, hate speech and offensive language has faced several key challenges in previous studies. Some of these challenges are:

- Most studies conflate hate speech and offensive language [25].
- Human coders do not pay attention to the context in which the slurs are used and might confuse offensive language with hate speech [25].

Table 7.1: Existing hate speech detection approaches

	supervised methods	syntactic/ PoS	grammatical features	meta-data	deep learning	web hyper-links	hate communities	H/O Differentiation
Burnap and Williams [18]	✓							N
Waseem and Hovy [102]	✓			✓				N
Xu and Zhu [103]			✓				✓	N
Gitari et al. [39]		✓						N
Silva et al. [92]		✓						N
Gambäck and Sikdar [36]	✓				✓			N
Chau and Xu [21]				✓		✓	✓	N
Ting et al. [97]							✓	N
Davidson et al. [25]	✓							Y

- There are very few published datasets to help train classification or embedding models.
- Bag-of-words approaches are not efficient since they tend to classify the textual data merely based on the presence of specific slurs and do not consider the context of the words [18, 25, 63].
- More advanced classification and embedding approaches might fail to differentiate hate speech from offensive language due to the scarcity of specific hateful comments and training data, e.g., hateful comments against specific nationalities on social networks [25].
- Users' meta data such as their ethnicity and gender can help detect hate speech, but this information is usually not available and is not reliable on social networks [25, 102].

7.1.3 Our Approach

None of the previous studies and methods address the moral aspect of abusive user-generated content. Therefore, there is a need to model this social phenomenon based on the insight provided by the moral values extracted from the social networks' textual data. Hate speech and offensive language contain moral weights, and using these weights can help analyze the moral statistics of these writings and use them as new means to automatically detect and differentiate them.

We believe a moral perspective can help better understand users' behavior when spreading hate speech and offensive language on social networks. In the following sections, we will study hate speech and offensive language utilizing a moral perspective.

7.2 Compute Tweets' Moral Loadings

The moral loadings are computed and utilized for document identification in a step-wise fashion described below:

1. **Represent tweets' vectors in the semantic space**: We employed the same scheme proposed in section 5.3, where we converted each tweet to a vector in the semantic space using Doc2Vec. We considered each tweet as a document and utilized each tweet as a separate paragraph to train the Doc2Vec PV-DM model. The tweets were represented in vectors of 400 in the semantic space. The window size used for training the Doc2Vec model was 1, and 5 words were used for negative sampling.

   We first trained the model. We then obtained the vectors' representation for each token in the corpus. For each tweet, we computed a vector, $t = (t_1, \ldots, t_{400})^T$, corresponding to the summation of the vectors of all the tokens in the tweet. Similarly, for each of the five moral foundations, we computed a vector, $f = (f_1, \ldots, f_{400})^T$, corresponding to the summation of the vectors of all the vice keywords in that moral foundation.

2. **Similarity-based moral loadings**: As described in section 5.3.3, the cosine similarity between t and f, which is the dot product between the two vectors, was used as the criterion for the similarity between each tweet and each moral foundation. Therefore, we define the moral loading m_{ij} for a tweet as the cosine similarity between the moral foundation f_i and the tweet t_j, where $i \in \{1,2,3,4,5\}$ is an index representing each moral foundation and j is a tweet's index for identification in the entire corpus.

3. **Classify/Label tweets based on moral foundations**: By applying our framework which was discussed in chapter 5, we calculate a moral loading for each tweet and a corresponding moral foundation. To identify tweets in each moral foundation, we assign each tweet to a dominant moral foundation with which the tweet has the highest cosine similarity/moral loading. Number of tweets in each foundation is shown in Table 7.2.

7.3 Tweets' Moral Statistics and Correlations

In this section, we present the statistics of hate speech and offensive language in our dataset. We discuss the frequency and conditional relative frequency of the tweets in each moral foundation for hate speech and offensive language, as well as the correlations between the moral foundations.

7.3.1 Frequency and Conditional Relative Frequency

Figure 7.1, shows frequency of the tweets in each moral foundation. We can see that most of the tweets are in the offensive language category since offensive language such as cursing in daily tweets is prevalent in social media [99].

However, the dataset is imbalanced with 19,190 tweets labeled as offensive language while only 1,430 tweets are labeled as hate speech and thus we present the conditional relative frequency in Figure 7.2, since due to this imbalance, we believe the conditional relative frequency is more reliable and can give us a better understanding of the moral break down. We utilize equation 5.1 to compute the CRF.

If we consider the hate speech and offensive language bars in Figure 7.2, two moral foundations: fairness and authority are in favor of hate speech while purity, ingroup, and harm are mostly about offensive language. This means that out of all tweets which were categorized as hate speech 25% are in ingroup and only less than 10% are in harm. While out of all tweets with the 'neither' label, more than 40% are in ingroup. In addition, we can see that fairness, ingroup, purity, and authority have a relatively similar share of 22% to 25% of the hate speech tweets while less than 10% of the 'hate speech' label was in harm.

7.3.2 Correlations between Moral Foundations

Next, we calculated the correlations between moral foundations using Pearson's r which was described in equation 5.2. In Table 7.3 for hate speech, all moral foundations are showing relatively high correlations. We observe that two binding moral foundations authority and ingroup are show-

Table 7.2: Number of tweets in each moral foundation

Moral Foundation	Hate Speech	Offensive Language	Neither
Fairness	327	3,682	837
Harm	131	2,653	186
Ingroup	338	4,765	1,774
Purity	326	4,984	284
Authority	308	3,106	1,118

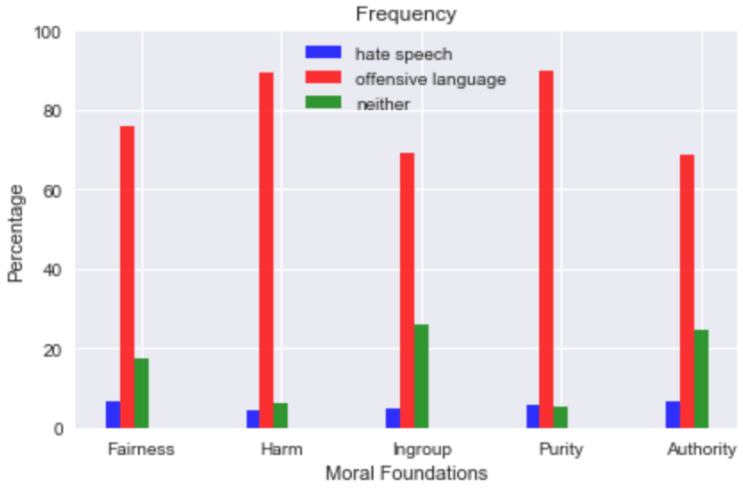

Figure 7.1: Frequency of the tweets in each moral foundation

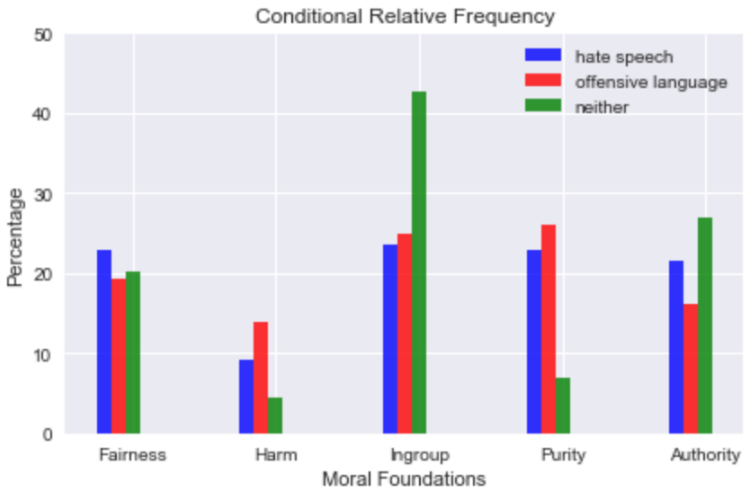

Figure 7.2: Conditional relative frequency of the labels

Table 7.3: Cosine similarities' correlations for hate speech

Moral Foundations	Fairness	Harm	Ingroup	Purity	Authority
Fairness	-	0.7579	0.8020	0.8169	0.85211
Harm	-	-	0. 7667	**0.7479**	0.7984
Ingroup	-	-	-	0.8099	**0.8581**
Purity	-	-	-	-	0.7793
Authority	-	-	-	-	-

Table 7.4: Cosine similarities' correlations for offensive language

Moral Foundations	Fairness	Harm	Ingroup	Purity	Authority
Fairness	-	0.7281	0.7402	0.7778	**0.8286**
Harm	-	-	0. 7319	**0.7138**	0.7835
Ingroup	-	-	-	0.7798	0.8282
Purity	-	-	-	-	0.7673
Authority	-	-	-	-	-

ing the highest correlation while the individual moral foundations harm and fairness are not as highly correlated as the binding moral foundations. Purity and ingroup which are binding moral foundations also have a comparatively high correlation.

As shown in Table 7.4, offensive language has relatively lower correlations between moral foundations. However, we can see that authority which is a binding moral foundation has the highest correlation with harm which is an individual moral foundation. We expected the correlation between individual foundations harm and fairness to be higher than authority and fairness, however, we did not observe this in our results.

Interestingly, authority and ingroup have the second highest correlation in offensive language while they had the highest correlation in hate speech as well.

Finally, we are now able to answer one of our initial questions which was if we can find a similar correlation pattern for two different datasets of different contexts, i.e., the Yelp dataset discussed in chapter 6 and chapter 7's hateful and offensive tweets. Authority and ingroup had a high correlation in chapter 6 as well. Therefore, authority and ingroup show a strong correlation disregarding the context in our studies. In addition, individual moral foundations harm and fairness did not show a strong correlation relative to other correlations in our datasets including the Yelp

reviews.

7.4 Enhance the Hate Speech Detection in Tweets

In this section, we will use the moral loadings as new features to improve our baseline approach for classifying "hate speech", "offensive language", and "neither" labels.

7.4.1 Baseline and the Improved Models

We will compare five models:

1. **Baseline** - The features used in the baseline approach are unigram, bigram, and trigrams weighted by their tf-idf score, part of speech tags which are unambiguous grammatical labels assigned to words in the context based on the words' roles in the sentence and computed using `NLTK` [95]. PoS tags capture the syntactic attributes of the data. We also incorporated binary and counts of the hashtags, mentions, retweets, URLs, features for the number of characters, words, and syllables in each tweet. A similar set of features was adopted in Davidson et al. [25]'s study on hate speech and offensive language detection.

2. **Baseline + doc vectors** - We used the document vectors from Doc2Vec as another set of features to compare their ability in capturing the semantic similarities to the baseline approach and to their five moral by-products. We added Doc2Vec features to the baseline approach for the second model.

3. **Baseline + doc vectors + moral loadings** - We added the five moral features to the second model. These five features are the moral loadings of each tweet which were extracted by the cosine similarity of the vector of the vice moral words and the tweets' vectors in the semantic space using Doc2Vec.

4. **Baseline + moral loadings** - In this model, we added the five moral features to the baseline model.

5. **Ensemble baseline + moral loadings** - We used the ensemble combination of an extremely randomized trees model, logistic regression with L2 regularization, and a gradient boosting model for the ensemble results based on the experimental performances of all tested algorithms. This model is similar to the fourth model; however, the classification algorithm is an ensemble of several models instead of utilizing one model.

   The gradient boosting method introduced by Jerome Friedman [34, 35] is a forward stepwise additive method that tries to strengthen a set of weak learners by optimizing different differentiable loss functions and implementing gradient descent. The algorithm we used was based on Friedman's mean squared error (MSE) which is computed as the mean squared error with improvement, and a deviance loss function used for probabilistic results [73].

7.4.2 Feature Selection

To reduce the dimensionality of the features we use an extremely randomized trees algorithm to detect the best features which usually has similar results to Random Forests but the training process can be faster [38]. Tree-based approaches are common ways for feature selection due to their easy implementation, robustness, consideration for non-linear correlations among features (where linear approaches fail), and implicit feature selection [15, 17]. However, using a tree-based approach enables us to do feature selection by looking at the selected features' importance and rankings with `scikit-learn` library [80]. This feature importance is based on Gini impurity which is a non-purity split approach, measuring how often a randomly chosen element would be incorrectly misclassified if it was randomly labeled based on class distributions [16]. This attribute helped us look through all features' score-rankings and find the importance of the moral features compared to the entire feature set. We could detect useful and ineffective features instead of choosing features uncritically. Moral features' rankings will be discussed in section 7.4.7.2.

All features defined in section 7.4.1 have undergone the feature selection step which means we do not introduce any new features to our models after doing feature selection, e.g., we add the moral loadings to the features in the baseline approach and then perform feature selection to select

the best set of features. We believe introducing features directly and without doing a selection is not reasonable since the feature selection step is the criterion for choosing the best sets of features for all corresponding models and thus we might introduce ineffectual new features by skipping the feature selection step and incorporating the new features directly in the models.

7.4.3 Split the Dataset

We randomly divide the data to an 8:1:1 ratio which are the training set, the validation set, and the test set, respectively. The results presented in Tables 7.5 and 7.6 are based on the test set's results. We will provide the accuracy for the test set and the validation set.

7.4.4 Classification Algorithm

We adopted several classification algorithms listed below and compared their performance based on the overall classification performance and the performance for each label:

- decision trees
- logistic regression (with L1 or L2 regularization)
- *k*NN
- Naive Bayes
- Random Forests
- LinearSVC

From our experiments, logistic regression with L2 regularization has the best performance. So, we used a logistic regression one-vs-all approach to classify the tweets. All the hyper parameters of the models were tuned on the validation set using a grid search to avoid over-fitting. In addition, as mentioned in section 7.4.1, we used the ensemble combination of an extremely randomized trees model, logistic regression with L2 regularization, and a gradient boosting model for the fifth model.

Table 7.5: Overall classification performance

model	Precision	Recall	F_1 Score	Validation Accuracy	Test Accuracy
1. Baseline	82	83	81	83.59	83.22
2. Baseline + doc vectors	83	84	81	85.30	83.66
3. Baseline + doc vectors + moral loadings	83	84	82	84.54	84.26
4. Baseline + moral loadings	84	84	82	84.63	84.31
5. Ensemble baseline + moral loadings	**87**	**88**	**87**	**89.24**	**87.90**

Table 7.6: Classification performance for each label

	Precision			Recall			F_1 Score		
model	Hate	Offensive	Neither	Hate	Offensive	Neither	Hate	Offensive	Neither
1. Baseline	65	83	93	27	97	76	38	89	84
2. Baseline + doc vectors	**70**	83	91	28	97	**79**	40	90	**85**
3. Baseline + doc vectors + moral loadings	69	83	92	29	97	**79**	40	90	**85**
4. Baseline + moral loadings	**70**	83	**94**	30	**98**	77	**42**	90	**85**
5. Ensemble baseline + moral loadings	46	**90**	93	**37**	96	77	41	**93**	84

7.4.5 Imbalance Handling

Since the data is overly imbalanced, we need a strategy to handle this imbalance. We used Synthetic Minority Oversampling TEchnique (SMOTE) in the training step of the classification to handle the imbalanced data since class-imbalanced data favors the majority class. The learner performs well for the majority class while the minority class is misclassified. This problem affects the true classification results and is more severe if we have a lot of features [48, 69].

The overall performances of the models are shown in Table 7.5. All classification tasks were implemented using the `scikit-learn` library in Python [80].

7.4.6 Performance Evaluation

Accuracy is an intuitive measure where we calculate the ratio of predictions which were accurate. While precision is the ratio of predicted positive observations which are correct over the count of total predicted positive observations. Recall is the fraction of correctly predicted positive observations to all observations in each class. Finally, F_1 score/F-measure is the weighted average of precision and recall to represent the performance with one value and takes both false positives and false negatives into consideration.

Overall performance - document vectors: As it is shown in Table 7.5, the baseline model has

82% precision while adding the document vectors improves the performance to 83%. Similarly, adding the document vectors to the baseline approach improves the recall from 83% to 84%. In addition, adding the document vectors improves the validation accuracy from 83.5% to 85.30%.

Overall performance - moral loadings: The moral loadings which are the moral by-products of the documents vectors combined with the baseline approach (fourth model) have the best performance among the first four models with 84% precision. The moral loadings and baseline approach (fourth model) can perform on par with the moral loadings combined with document vectors and the baseline approach (third model) with 84% and 82% for recall and F-measure, respectively. Moreover, the validation and test accuracy for baseline approach combined with moral loadings (fourth model) are the best among the first four models.

The ensemble approach with moral loadings and the baseline approach (fifth model), clearly has the best results among all five models with 87% precision and F-measure, 88% recall, more than 89% validation accuracy, and nearly 88% test accuracy.

Each label's performance - document vectors: Table 7.6 lists the performance for each label. We observe that adding the document vectors or their moral by-products improves the performance. For instance, adding the document vectors improves hate precision from 65% to 70% while we see a similar performance among all models for offensive's precision with 83%. Similarly, adding the moral loadings/document vectors improves the recall for hate speech from 27% to 28% and neither's recall form 76% to 79%. Moreover, adding document vectors improves hate' F-measure to 40%, offensive's F-measure to 90%, and neither's F-measure improves by 1% by adding the document vectors and/or moral loadings. Interestingly, we observe a precision-recall trade-off for 'neither'. A better recall means sacrificing precision (and contrariwise) [104].

Each label's performance - moral loadings: The fourth model improves hate's precision to 70%, while the third model also improves the baseline approach to 69%. In addition, the baseline approach combined with moral loadings (fourth model) has the best precision for neither with 94%. While adding the document vectors reduces neither label's precision from 93% to 92%. The fourth model improves hate's recall from 27% to 30%. Similarly, the third model improves the baseline

approach's performance to 29%. The fourth model is the only model that improves offensive's recall. In all instances, the third and fourth model have improved the F_1 score. Hate speech's F_1 score improves significantly in the fourth model (from 38% to 42%) by combining the moral loadings with the baseline approach.

The fifth model performs best for offensive's precision and recall with 90% and 93%, respectively. It also improves the hate recall to 37%.

7.4.7 Discussions

In this section, we will discuss the results and the reasons behind the performance of the improved models.

7.4.7.1 Moral Features' Performance

As shown in Table 7.5, the five moral features which are the moral loadings of the tweets are improving the baseline approach. This improvement is the justification that the moral loadings of hate speech and offensive language can help classify and distinguish the two. Adding the moral loadings alone (fourth model) has a better performance than adding document vectors or these features and moral loadings at the same time. Therefore, using the moral foundations dictionary and finding the moral loadings of the tweets is the best approach to detect hate speech and offensive language. This model performs better than simply adding document vectors or introducing an excessive number of features by adding both Doc2Vec features and moral loadings.

In Table 7.6, we see that using moral loadings helps improving the precision, recall, and F_1 score of hate speech. Moral features are performing well for offensive language and neither classes as well, with neither class's recall as the only exception. In general, the baseline approach combined with the moral loadings (fourth model) has the best performance for hate/neither precision, offensive recall, and hate/neither F-measure. Moreover, the ensemble model (fifth model) has the same set of features as the fourth model combined with an improved/ensemble classification algorithm. Therefore, we can count offensive's precision and F_1 score, and hate recall in this category.

The moral features combined with the baseline (fourth model) without the document vectors are performing better due to two reasons:

1. We are adding an excessive number of features by incorporating document vectors and moral loadings which results in identifying every single data point by single features and generating a special case for each data point, i.e., the more features we add, the larger the hypotheses set will be; however, this approach results in a poor performance for the test set. This phenomenon is also referred to as the "curse of dimensionality" for high-dimensional data [11, 12, 31].

2. Moral features are the by-products of the document vectors and thus capture their semantics. Adding the document vectors and moral features is redundant and reduces the effect of the moral loadings on the prediction given a larger set of hypotheses.

7.4.7.2 Moral Features' Rankings

The rankings of the moral features from the first step which was feature selection, also shows that the moral features are highly important with authority ranked 5, ingroup ranked 9, fairness ranked 17, purity ranked 38, and harm ranked 86 among 11,165 features based on their Gini measure for the best-performing *baseline + moral loadings* (fourth) model. Higher rankings imply more important features. These rankings go along with the classification performance of the moral loadings which improved the baseline approach 's overall classification performance and the performance for each label. Interestingly, harm, which had the lowest share of hate speech, offensive language, and neither labels in Figure 7.2, has the lowest ranking among all moral features.

Moreover, ingroup and authority are of two highest rankings among all moral foundations. This is in line with our correlation results, since the correlation between authority and ingroup was high in offensive and hateful tweets, as well as the Yelp reviews.

7.4.7.3 Hate Speech's Low Performance

Davidson et al. [25] emphasized that most of the misclassification is occurring in hate speech.

There are several reasons behind this phenomenon according to their study and our results:

- The first issue is that the human coders mostly labeled tweets with bold homophobic and racial slurs as hate speech while they did not label true hate speech tweets without bold slurs correctly. If the tweet was hateful but did not contain any slurs at all, the coders mostly labeled it as neither. So this low performance is partly due to the coders lack of attention to the context in which slurs were/were not used; although, there are some true misclassified labels [25].

- The second issue, which was also mentioned in Davidson et al. [25], is some hateful posts are rare among typical hate speech tweets. For instance, we have several anti-black tweets where the classifier performs well in detecting their labels; however, we have rare cases of hateful tweets against other nations/specific nationalities where the classifier does not have enough training data to detect them correctly.

- Moreover, the classifiers sometimes fail to distinguish the slurs that are used in a daily conversation context and the ones that are truly hateful and targeting victims. One major contribution of differentiating hate speech and offensive language is the classifiers' high performance in correctly detecting offensive language. Conflating these two languages results in labeling people who are merely using slurs in their daily lives as hate speakers, which might entail legal prohibitions. It is true that combining the offensive and hateful tweets and labeling them as hate speech, similar to a multitude of previous studies, will drastically improve the performance; however, this approach will not fulfill our goal to differentiate these two.

- Despite the key challenges discussed earlier, using the moral loadings improves the classification performance of hate speech. In the overall performance in Table 7.5, the validation and test accuracy, which are the direct metrics of intuitive misclassification, have improved by nearly 1%. In addition, the overall precision has improved from 82% to 84%. The overall recall and F_1 score have improved as well. In Table 7.6, we can see that using the moral loadings improved hate speech's precision from 65% to 70%, its recall has improved from

27% to 30%, and its F_1 score has improved from 38% to 42%. Therefore, using the moral features helps enhancing hate speech prediction performance.

7.4.7.4 Ensemble Model's Precision and Recall

In our fifth model, we used an ensemble model to further improve the baseline approach. The ensemble method has improved the overall result since we are using the baseline features combined with moral loadings (fourth model) which had the best performance among the first four models and further enhancing its prediction by combining several classification algorithms and voting among three models. However, this improvement is mostly contributed to offensive language's significant improvement in precision. There are two reasons behind the sudden drop in hate speech precision:

1. The ensemble model is sacrificing the hate speech precision, which drops to 46%, for its recall. High recall means hurting precision (and contrariwise) [104].

2. Using an ensemble approach rewards the overall performance due to the voting approach among several models; however, it penalizes the hate precision because of incorporating the extremely randomized tree in the fifth model. If a single extremely randomized tree is adopted to classify the baseline approach, the performance will be 41%, 36%, and 38% for precision, recall, and F_1 score respectively. These results imply that a tree-based approach has a lower precision, but higher recall compared to the first four models in Table 7.6. We can observe the same pattern for the ensemble approach in Table 7.6. The ensemble approach has a lower precision and higher recall for hate speech compared to the other four models. Therefore, incorporating the tree-based in the ensemble model helps the overall performance, however, it penalizes hate's precision due to its inability to classify the hate label.

   In addition, the tree approach does not have a good performance for the neither class. The baseline approach for 'neither' classified with a single extremely randomized model has 67%, 73%, and 70% performances for precision, recall, and F_1 score, respectively. So the

ensemble model which incorporates the tree-based approach does not show an improvement for this class relative to other models.

One can use manual weighting schemes for the models in an ensemble to set a threshold to handle the infamous precision-recall trade-off or reduce a model's strength in the voting process.

7.4.7.5 Aggregation of Tf-idf and Doc2Vec for Very Short Texts

In this section, we will discuss the reasons behind combining document vectors with tf-idf features and why they outperform the baseline approach of the traditional tf-idf features.

De Boom et al. [26], mention that the textual data available in social media is mostly of very short texts. In particular, each tweet is of approximately thirty words. Tf-idf as the traditional method in the text mining community, works well when we have more word overlaps and its best performance is for a document of length 30 words. However, sometimes social media texts can be shorter than 30 words which means we will not have enough word overlap for tf-idf. This is where word embeddings such as Doc2Vec that can capture the semantic similarities between words can be useful [26, 74, 75, 76]. De Boom et al. [26] performed further toy tests to justify that adding the semantic information from a word embedding approach to the traditional tf-idf will improve the performance of tf-idf while for longer sentences, e.g., tweets of length 30 words, tf-idf alone can produce comparable results. In another study of De Boom et al. [27], the authors focused on tweets as their main social media textual data and aggregated the traditional tf-idf with the word embeddings' vectors using a loss function to optimize the performance. All combined approaches outperformed the tf-idf baseline approach significantly due to their ability to capture the semantic similarities.

In conclusion, we observe the same pattern in our results. Our performance has improved by adding the document vectors due to the short texts available in our tweets. In addition, the moral loadings which are the by-products of our document vectors, combined with the baseline approach (fourth model) is our best-performance model since the tweets in our dataset are of various lengths:

from one word to longer sentences close to 30 words. By combining the tf-idf features with the moral loadings obtained from Doc2Vec, we are adding two layers of information to the baseline approach:

1. We add the semantic similarities which help improve the performance for shorter tweets where tf-idf fails.

2. We are adding the moral weights of the tweet by using the moral loadings as features.

7.5 Summary

In this chapter, we studied hate speech and offensive language based on tweets' moral loadings. We used a crowd-sourced manually labeled dataset which labeled the tweets as offensive language, hate speech, and neither. We employed the moral foundations theory to analyze the moral break down for the three classes. We used Doc2Vec to represent the tweets in the semantic space and defined the cosine similarity of the average of the moral vice key words in the moral foundations dictionary for each moral foundation and the tweets' vectors as the measure of similarity to each moral foundation. This similarity was called tweets' moral loadings. To identify tweets in each moral foundation, we assigned the tweets to a moral foundation with which the tweets had the highest similarity.

We first discussed the frequency and conditional frequency of the tweets. We observed that purity, ingroup, and harm are mostly offensive language while fairness and authority are in favor of hate speech (only comparing the hate speech and offensive labels). Furthermore, we noticed that in general, moral foundations in hate speech have higher correlations compared to offensive language. Moreover, binding moral foundations were of higher correlations in hate speech compared to offensive language.

We then studied the performance of moral features to predict social media data. In particular, we classified the tweets by creating a baseline approach of tf-idf features, PoS tags, and several other features such as counts of retweets, number of characters, words, etc. We added the document

vectors and the moral loadings to the baseline features to justify that moral loadings are effective in social networks' data classification and prediction, e.g., the hateful and offensive tweets. Our results show that the moral loadings can improve the baseline approach and can help the misclassification of the hate label which was a key challenge in previous studies.

Chapter 8

Conclusions and Future Work

In this work, we argue that text mining is an important tool to extract knowledge from the textual data on social networks. However, social networking data analysis/mining and NLP techniques are facing several challenges such as dealing with the abundance of data and domains or the quantification of the moral and social values of the data. We can tackle these challenges by using new boosting features.

We argue that our daily activities including our actions in social networks are bound with morality and culture to certain extent. Therefore, we propose a novel approach to study social networks' data based on moral features in specific domains such as business rating platforms and social networks' abusive content.

To analyze this relationship and to justify the importance of a moral approach in the presence of concrete moral patterns on social networks, we studied the influence of moral foundations on reviewers' rating behavior in the immorality context. We performed a similar study for immorality based on hate speech and offensive language in social networks.

- We used the moral foundations proposed in the Moral Foundations Theory (MFT) [46, 47] and the vice keywords defined in the Moral Foundations Dictionary (MFD) to support a natural language processing analysis on a dataset of online reviews from the Yelp Challenge.

- We adopted the word embedding method, Doc2Vec, to convert the reviews to vector repre-

sentations in the semantic space defined by the reviews. By converting the reviews and the MFD keywords for each moral foundation into vectors, we can calculate the moral loading of the review in the form of cosine similarity between the vector of the document and the vector of the average vice words.

- Using an experimentally defined threshold for cosine similarities, we identified a moral corpus and a corresponding moral-concerned user set for each moral foundation.

- We investigated the frequency and conditional relative frequency of review ratings for the overall moral corpus and the moral corpora associated with each of the five moral foundations, as well as the rating distributions of the regular users who rated the same set of businesses. The comparison shows that the rating pattern of regular users differs significantly from the one of the moral-concerned users. Moreover, our findings indicate that people with moral concerns tend to rate lower if a moral foundation is violated. CRF was one of our two metrics to study the moral features' importance in social networks.

- For the moral-concerned users, we also studied differences, in terms of the average rating and the weighted average ratings, between their moral-related reviews and their all reviews with no moral consideration. Results in the cumulative density functions reveal that there is a higher likelihood of a smaller difference if we consider each reviewer's moral loading. Moreover, moral-concerned users tend to elicit the same moral tendencies in their general average rating.

- Next, we examined the correlations between the moral foundations themselves as the second metric to understand morality on social networks. Purity was shown to be the most distinctive moral foundation.

In our second study, we addressed the problem of user-generated abusive content in social networks as a second means for analyzing the importance and the break down of morality in social networks. In particular, we studied hate speech and offensive language and their difference which

lies in the manner they are used. Some curse words are simply part of people's daily conversations and their daily tweets [99, 101]. However, some speech are targeted towards minority groups based on their natural features [25, 39].

We argued that hate speech in itself has an obvious moral weight and a moral study can help detect and differentiate hate speech and offensive language. We used a dataset of hateful and offensive tweets to analyze the moral features' performance in the prediction of social media data.

- To perform our study, we used the concepts we had previously used, i.e., we used moral foundations dictionary's vice keywords and employed Doc2Vec to represent the tweets in the semantic space. We called the cosine similarity of the average of the moral keywords for each moral foundation with each tweets' vector, the moral loading of the tweet with respect to that moral foundation. Next, we assigned each tweet to a dominant moral foundation with which it had the highest similarity.

- We presented the frequency and conditional relative frequency of the tweets. Our results show that most tweets are offensive since slurs are prevalent in social media [99]. However, in our conditional relative frequency, we observed a similar share of 22% to 25% of the tweets labeled as hate speech for fairness, ingroup, purity, and authority. While more than 40% of the tweets labeled as 'neither' are in ingroup.

- We studied the correlations between the moral foundations as a second measure of understanding morality in social networks. We observed that in general, hate speech had higher moral correlations and there were higher correlations between 'binding' moral foundations. Offensive language showed a high correlation between fairness, an individual moral foundation, and authority, a binding moral foundation.

- We observed a high correlation between authority and ingroup in the hate speech, offensive language, and Yelp datasets.

- Next, we used the moral loadings as new features in a multi-class classification to improve the classification performance of a baseline approach of tf-idf features, PoS tags, and several

additional features such as counts of retweets, number of characters, syllables, etc. We used an extremely randomized trees algorithm for reducing the dimensionality of data and feature selection and we classified the tweets using logistic regression with L2 regularization after examining several classification algorithms. Our results indicated that adding the five moral loadings improved the general baseline model in terms of precision, recall, F-measure, and test/validation accuracy. Moreover, our moral features had higher rankings based on their Gini impurity index compared to the entire feature set.

- Lastly, we stated the reasons behind the performance of our models for very short texts which are prevalent on social media [26]. Adding embedding features such as Doc2Vec features, can improve the performance since it can capture the semantic similarities for sentences shorter than 30 words [26, 74, 75, 76]. By adding the moral features we are adding this semantic information and a layer of moral information and thus we observe an improved performance.

We believe this work represents a new avenue of analysis on moral psychology and online social networks. For future work, we will test our findings with larger corpora and more word embedding methods. We will also consider making a comprehensive dictionary to filter the moral reviews. In addition, we anticipate developing an extended version of MFD, which incorporates more words in the same medium. We also hope to incorporate emotion recognition and sentiment analysis in this study to improve the results. Finally, we hope to incorporate this study in other abusive behavior in social media such as cyberbullying.

References

[1] CrowdFlower. https://www.crowdflower.com/.

[2] Moral Foundations Dictionary. http://moralfoundations.org/.

[3] Twitter's number of active users. https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/.

[4] . Yelp Dataset Challenge. https://www.yelp.com/dataset/challenge.

[5] . Yelp fact sheet. https://www.yelp.com/factsheet.

[6] M. Adedoyin-Olowe, M. M. Gaber, and F. Stahl. A survey of data mining techniques for social media analysis. *arXiv preprint arXiv:1312.4617*, 2013.

[7] A. Agarwal, B. Xie, I. Vovsha, O. Rambow, and R. Passonneau. Sentiment analysis of twitter data. In *Proceedings of the workshop on languages in social media*, pages 30–38. Association for Computational Linguistics, 2011.

[8] M. Al Hasan and M. J. Zaki. A survey of link prediction in social networks. In *Social network data analytics*, pages 243–275. Springer, 2011.

[9] A. Barrón-Cedeno, P. Rosso, E. Agirre, and G. Labaka. Plagiarism detection across distant language pairs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 37–45. Association for Computational Linguistics, 2010.

[10] B. Batrinca and P. C. Treleaven. Social media analytics: a survey of techniques, tools and platforms. *Ai & Society*, 30(1):89–116, 2015.

[11] R. Bellman. Dynamic programming (dp). 1957.

[12] R. E. Bellman. *Adaptive control processes: a guided tour*. Princeton university press, 2015.

[13] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[14] R. L. Boyd, S. R. Wilson, J. W. Pennebaker, M. Kosinski, D. J. Stillwell, and R. Mihalcea. Values in words: Using language to evaluate and understand personal values. In *ICWSM*, pages 31–40, 2015.

[15] L. Breiman. Consistency for a simple model of random forests. 2004.

[16] L. Breiman. *Classification and regression trees*. Routledge, 2017.

[17] L. Breiman and A. Cutler. Random forests-classification description. *Department of Statistics, Berkeley*, 2, 2007.

[18] P. Burnap and M. L. Williams. Cyber hate speech on twitter: An application of machine classification and statistical modeling for policy and decision making. *Policy & Internet*, 7 (2):223–242, 2015.

[19] M. Campr and K. Ježek. Comparing semantic models for evaluating automatic document summarization. In *International Conference on Text, Speech, and Dialogue*, pages 252–260. Springer, 2015.

[20] P. J. Carrington, J. Scott, and S. Wasserman. *Models and methods in social network analysis*, volume 28. Cambridge university press, 2005.

[21] M. Chau and J. Xu. Mining communities and their relationships in blogs: A study of online hate groups. *International Journal of Human-Computer Studies*, 65(1):57–70, 2007.

[22] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography, proceedings of the 27th annual meeting of the association for computational linguistics. 1989.

[23] K. W. Church and P. Hanks. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29, 1990.

[24] C. J. Collins and K. D. Clark. Strategic human resource practices, top management team social networks, and firm performance: The role of human resource practices in creating organizational competitive advantage. *Academy of management Journal*, 46(6):740–751, 2003.

[25] T. Davidson, D. Warmsley, M. Macy, and I. Weber. Automated hate speech detection and the problem of offensive language. *arXiv preprint arXiv:1703.04009*, 2017.

[26] C. De Boom, S. Van Canneyt, S. Bohez, T. Demeester, and B. Dhoedt. Learning semantic similarity for very short texts. In *Data Mining Workshop (ICDMW), 2015 IEEE International Conference on*, pages 1229–1234. IEEE, 2015.

[27] C. De Boom, S. Van Canneyt, T. Demeester, and B. Dhoedt. Learning representations for tweets through word embeddings. In *Benelearn*, 2016.

[28] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6): 391, 1990.

[29] M. Dehghani, K. Johnson, J. Hoover, E. Sagi, J. Garten, N. J. Parmar, S. Vaisey, R. Iliev, and J. Graham. Purity homophily in social networks. *Journal of Experimental Psychology: General*, 145(3):366, 2016.

[30] L. R. Dice. Measures of the amount of ecologic association between species. *Ecology*, 26 (3):297–302, 1945.

[31] D. L. Donoho et al. High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture*, 1:32, 2000.

[32] N. Evangelopoulos, X. Zhang, and V. R. Prybutok. Latent semantic analysis: five methodological recommendations. *European Journal of Information Systems*, 21(1):70–86, 2012.

[33] L. C. Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1 (3):215–239, 1978.

[34] J. H. Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[35] J. H. Friedman. Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38 (4):367–378, 2002.

[36] B. Gambäck and U. K. Sikdar. Using convolutional neural networks to classify hate-speech. In *Proceedings of the First Workshop on Abusive Language Online*, pages 85–90, 2017.

[37] J. Garten, R. Boghrati, J. Hoover, K. M. Johnson, and M. Dehghani. Morality between the lines: Detecting moral sentiment in text. In *Proceedings of IJCAI 2016 workshop on Computational Modeling of Attitudes, New York, NY. Retrieved from http://mortezadehghani. net/wp-content/uploads/morality-lines-detecting. pdf*, 2016.

[38] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine learning*, 63 (1):3–42, 2006.

[39] N. D. Gitari, Z. Zuping, H. Damien, and J. Long. A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering*, 10(4):215–230, 2015.

[40] Y. Goldberg and O. Levy. word2vec explained: Deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*, 2014.

[41] W. H. Gomaa and A. A. Fahmy. A survey of text similarity approaches. *International Journal of Computer Applications*, 68(13), 2013.

[42] L. A. Goodman and W. H. Kruskal. Measures of association for cross classifications. *Journal of the American statistical association*, 49(268):732–764, 1954.

[43] J. Graham, J. Haidt, and B. A. Nosek. Liberals and conservatives rely on different sets of moral foundations. *Journal of personality and social psychology*, 96(5):1029, 2009.

[44] J. Haidt. The positive emotion of elevation. 2000.

[45] J. Haidt. Morality. *Perspectives on psychological science*, 3(1):65–72, 2008.

[46] J. Haidt. *The righteous mind: Why good people are divided by politics and religion*. Vintage, 2012.

[47] J. Haidt and C. Joseph. Intuitive ethics: How innately prepared intuitions generate culturally variable virtues. *Daedalus*, 133(4):55–66, 2004.

[48] H. Han, W.-Y. Wang, and B.-H. Mao. Borderline-smote: a new over-sampling method in imbalanced data sets learning. In *International Conference on Intelligent Computing*, pages 878–887. Springer, 2005.

[49] Z. S. Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

[50] T. B. Hashimoto, D. Alvarez-Melis, and T. S. Jaakkola. Word embeddings as metric recovery in semantic spaces. *Transactions of the Association for Computational Linguistics*, 4: 273–286, 2016.

[51] W. He, S. Zha, and L. Li. Social media competitive analysis and text mining: A case study in the pizza industry. *International Journal of Information Management*, 33(3):464–472, 2013.

[52] J. Hoover. Into the wild: Big data analytics in moral psychology. *structure*, 7(3):269–279.

[53] A. Hotho, S. Staab, and G. Stumme. Ontologies improve text document clustering. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 541–544. IEEE, 2003.

[54] X. Hu and H. Liu. Text analytics in social media. In *Mining text data*, pages 385–414. Springer, 2012.

[55] P. Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, 37:547–579, 1901.

[56] D. Jurafsky and J. H. Martin. *Speech and language processing*, volume 3. Pearson London:, 2014.

[57] A. Kao and S. R. Poteet. *Natural language processing and text mining*. Springer Science & Business Media, 2007.

[58] R. Kaur and K. Sasahara. Quantifying moral foundations from various topics on twitter conversations. In *Big Data (Big Data), 2016 IEEE International Conference on*, pages 2505–2512. IEEE, 2016.

[59] M. G. Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[60] S. Kim, J. Bak, and A. H. Oh. Do you feel what i feel? social aspects of emotions in twitter conversations. In *ICWSM*, 2012.

[61] E. F. Krause. *Taxicab geometry: An adventure in non-Euclidean geometry*. Courier Corporation, 1975.

[62] W. H. Kruskal. Ordinal measures of association. *Journal of the American Statistical Association*, 53(284):814–861, 1958.

[63] I. Kwok and Y. Wang. Locate the hate: Detecting tweets against blacks. In *AAAI*, 2013.

[64] T. K. Landauer and S. T. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological review*, 104(2):211, 1997.

[65] H. Landmann and U. Hess. Testing moral foundation theory: Are specific moral emotions elicited by specific moral transgressions? *Journal of Moral Education*, 47(1):34–47, 2018.

[66] J. H. Lau and T. Baldwin. An empirical evaluation of doc2vec with practical insights into document embedding generation. *arXiv preprint arXiv:1607.05368*, 2016.

[67] Q. Le and T. Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196, 2014.

[68] A. Lehman. *JMP for basic univariate and multivariate statistics: a step-by-step guide*. SAS Institute, 2005.

[69] G. Lemaître, F. Nogueira, and C. K. Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. *Journal of Machine Learning Research*, 18(17):1–5, 2017. URL `http://jmlr.org/papers/v18/16-365.html`.

[70] C. X. Lin, B. Zhao, Q. Mei, and J. Han. Pet: a statistical model for popular events tracking in social communities. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 929–938. ACM, 2010.

[71] C. Manning. D, raghavan, p, schütze, h,(2009). *An Introduction to Information Retrieval*.

[72] J. H. Martin and D. Jurafsky. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson/Prentice Hall, 2009.

[73] A. Mayr, H. Binder, O. Gefeller, and M. Schmid. The evolution of boosting algorithms. *Methods of information in medicine*, 53(06):419–427, 2014.

[74] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[75] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[76] T. Mikolov, W.-t. Yih, and G. Zweig. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, 2013.

[77] S. M. Mohammad, S. Kiritchenko, and X. Zhu. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. *arXiv preprint arXiv:1308.6242*, 2013.

[78] P. Nokhiz and F. Li. Understanding rating behavior based on moral foundations: The case of yelp reviews. In *Big Data (Big Data), 2017 IEEE International Conference on*, pages 3938–3945. IEEE, 2017.

[79] K. Pearson. Note on regression and inheritance in the case of two parents. *Proceedings of the Royal Society of London*, 58:240–242, 1895.

[80] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[81] J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[82] P. Penumatsa, M. Ventura, A. C. Graesser, M. Louwerse, X. Hu, Z. Cai, and D. R. Franceschetti. The right threshold value: What is the right threshold of cosine measure

when using latent semantic analysis for evaluating student answers? *International Journal on Artificial Intelligence Tools*, 15(05):767–777, 2006.

[83] R. Rehurek and P. Sojka. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer, 2010.

[84] C. Richthammer, M. Netter, M. Riesner, J. Sänger, and G. Pernul. Taxonomy of social network data types. *EURASIP Journal on Information Security*, 2014(1):11, 2014.

[85] P. Rozin, J. Haidt, and C. R. McCauley. Disgust. 1993.

[86] P. Rozin, L. Lowery, S. Imada, and J. Haidt. The cad triad hypothesis: a mapping between three moral emotions (contempt, anger, disgust) and three moral codes (community, autonomy, divinity). *Journal of personality and social psychology*, 76(4):574, 1999.

[87] E. Sagi and M. Dehghani. Measuring moral rhetoric in text. *Social science computer review*, 32(2):132–144, 2014.

[88] E. Sagi and M. Dehghani. Moral rhetoric in twitter: A case study of the us federal shutdown of 2013. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 36, 2014.

[89] H. M. Saleem, K. P. Dillon, S. Benesch, and D. Ruths. A web of hate: Tackling hateful speech in online social spaces. *arXiv preprint arXiv:1709.10159*, 2017.

[90] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information processing & management*, 24(5):513–523, 1988.

[91] G. Salton and J. Michael. Mcgill. 1983. *Introduction to modern information retrieval*, 1983.

[92] L. A. Silva, M. Mondal, D. Correa, F. Benevenuto, and I. Weber. Analyzing the targets of hate in online social media. In *ICWSM*, pages 687–690, 2016.

[93] C. Smith. *Moral, believing animals: Human personhood and culture*. Oxford University Press, 2003.

[94] S. Somasundaran and J. Wiebe. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 116–124. Association for Computational Linguistics, 2010.

[95] B. Steven, E. Klein, and E. Loper. Natural language processing with python. *OReilly Media Inc*, 2009.

[96] K. Subrahmanyam, S. M. Reich, N. Waechter, and G. Espinoza. Online and offline social networks: Use of social networking sites by emerging adults. *Journal of applied developmental psychology*, 29(6):420–433, 2008.

[97] I.-H. Ting, S.-L. Wang, H.-M. Chi, and J.-S. Wu. Content matters: A study of hate groups detection based on social networks analysis and web mining. In *Advances in Social Networks Analysis and Mining (ASONAM), 2013 IEEE/ACM International Conference on*, pages 1196–1201. IEEE, 2013.

[98] P. D. Turney and P. Pantel. From frequency to meaning: Vector space models of semantics. *Journal of artificial intelligence research*, 37:141–188, 2010.

[99] W. Wang, L. Chen, K. Thirunarayan, and A. P. Sheth. Cursing in english on twitter. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 415–425. ACM, 2014.

[100] X. Wang, L. Tang, H. Gao, and H. Liu. Discovering overlapping groups in social media. In *Data Mining (ICDM), 2010 IEEE 10th International Conference on*, pages 569–578. IEEE, 2010.

[101] W. Warner and J. Hirschberg. Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26. Association for Computational Linguistics, 2012.

[102] Z. Waseem and D. Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93, 2016.

[103] Z. Xu and S. Zhu. Filtering offensive language in online communities using grammatical relations. In *Proceedings of the Seventh Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*, pages 1–10, 2010.

[104] Y. Yang. An evaluation of statistical approaches to text categorization. *Information retrieval*, 1(1-2):69–90, 1999.

[105] A. X. Zhang and S. Counts. Modeling ideology and predicting policy change with social media: Case of same-sex marriage. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 2603–2612. ACM, 2015.