Stance Detection and Analysis in Social Media

by

Parinaz Sobhani

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements
For the Ph.D. degree in
Computer Science

School of Electrical Engineering and Computer Science
Faculty of Engineering
University of Ottawa

Abstract

Computational approaches to opinion mining have mostly focused on polarity detection of product reviews by classifying the given text as positive, negative or neutral. While, there is less effort in the direction of socio-political opinion mining to determine favorability towards given targets of interest, particularly for social media data like news comments and tweets. In this research, we explore the task of automatically determining from the text whether the author of the text is in favor of, against, or neutral towards a proposition or target. The target may be a person, an organization, a government policy, a movement, a product, etc. Moreover, we are interested in detecting the reasons behind authors' positions.

This thesis is organized into three main parts: the first part on Twitter stance detection and interaction of stance and sentiment labels, the second part on detecting stance and the reasons behind it in online news comments, and the third part on multi-target stance classification.

One may express favor (or disfavor) towards a target by using positive or negative language. Here, for the first time, we present a dataset of tweets annotated for whether the tweeter is in favor of or against pre-chosen targets, as well as for sentiment. These targets may or may not be referred to in the tweets, and they may or may not be the target of opinion in the tweets. We develop a simple stance detection system that outperforms all 19 teams that participated in a recent shared task competition on the same dataset (SemEval-2016 Task #6). Additionally, access to both stance and sentiment annotations allows us to conduct several experiments to tease out their interactions.

Next, we proposed a novel framework for joint learning of stance and reasons behind it. This framework relies on topic modeling. Unlike other machine learning approaches for argument tagging which often require a large set of labeled data, our approach is minimally supervised. The extracted arguments are subsequently employed for stance classification. Furthermore, we create and make available the first dataset of online news comments manually annotated for stance and arguments. Experiments on this dataset demonstrate the benefits of using topic modeling, particularly Non-Negative Matrix Factorization, for argument detection.

Previous models for stance classification often treat each target independently, ignoring the potential (sometimes very strong) dependency that could exist among targets. However, in many applications, there exist natural dependencies among targets. In this research, we relieve such independence assumptions in order to jointly model the stance expressed towards multiple targets. We present a new dataset that we built for this task and make it publicly available. Next, we show that an attention-based encoder-decoder framework is very effective for this problem, outperforming several alternatives that jointly learn dependent subjectivity through cascading classification or multi-task learning.

Acknowledgements

I would like to express my sincere thanks to my supervisors Professor Stan Matwin and Professor Diana Inkpen for their inspiring guidance and encouraging me to explore and develop ideas. It was a joy and a privilege to work with you. My Ph.D. days will be missed but I am deeply grateful to have had the opportunity to work with you.

I am very grateful and feel fortunate to work with scientists from the National Research Council of Canada. In particular, I am indebted to Xiaodan Zhu who took me to the world of deep neural networks without the support and motivation from whom I would never have started exploring this new world. My gratitude also goes to Saif Mohammad and Svetlana Kiritchenko, whom I have learned a lot from and helped me to move and shape this research in a right direction.

This journey would have never been possible if not for the generous help and unfailing support and love of my husband Adel. Thanks for always having faith in me and encouraging me to face challenges with enthusiasm.

My deepest appreciation goes to my father Abbas and my mother Batoul for their endless love and support. I owe you the constant desire that I have to aim high, to achieve big, and to strive for better. Many thanks to my sisters Parisa and Farahnaz for their constant source of love, concern, support and strength all these years.

Thanks to my friends Rahil and Ebrahim for your wonderful companion. I would never forget your help and those beautiful moments I shared with you. I thank Elnaz Bigdeli for her sound advice and support since my first days in Ottawa. Thanks to my lovely friends Hoda and Nikta whose love and support far away from here is always with me.

Thanks to all TAMALE group members for discussions and their thoughtful comments and feedback. To the human annotators that have contributed in creating the dataset as part of this research.

Last but not least, many thanks to the members of my examining committees – Professors John Oommen, Nathalie Japkowics and Herna Viktor for their insightful, wise comments. I thank Professor Claire Cardie for her time as the external examiner of this thesis and giving me valuable feedback.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Overview

Nowadays, the Web is considered as a source of opinions of online users. These opinions are a valuable source of feedback about products, current issues, events, etc. They provide an exceptional opportunity for decision makers to deploy these sources of data to identify public opinions about their target of interest. For instance, policy makers who are interested in determining citizens reaction to new policies may use posts in forums, Facebook, and Twitter related to their target policy in order to collect feedback. The main challenge is the tremendous amount of such textual data that makes their manual analysis time-consuming and costly.

Efforts for automatic text understanding and analysis are part of the Natural Language Processing (NLP) research area. The majority of NLP methods are based on statistical methods inspired from various fields such as machine learning (ML), data mining, and information retrieval. Sentiment analysis and stance detection can be considered as transitional steps towards text understanding. We believe that stance classification goes one step further than sentiment analysis as it is more complicated than classifying text into expressing a positive or a negative opinion.

In this thesis, our main objective is to explore stance detection in social media texts, particularly tweets and news comments. Stance detection is the task of automatically determining from the text whether the author of the text is in favor of or against of a target of interest. Automatically detecting stance has widespread applications in information extraction, text summarization, and textual entailment. Recently, stance detection has been widely used in fake news detection. Specifically, the task is formulated as detecting the stance toward the news headline for different text spans of the news article to detect contradictions.

1.2 Motivation

In the past, people were only consumers of information on the Web. With the advent of Web 2.0, new tools for producing User Generated Content (UGC) were provided. Consequently, huge amounts of data are generated every day on the Web. Most of these data are in form of text and are originally meant for human consumption and communication. Currently, there are increasing amounts of text on the Web in forms of product reviews, blogs, forums, and social media posts on platforms such as Twitter and Facebook.

As the volume of such unstructured data increased, the request for automatic identification and extraction of opinions grew significantly. Furthermore, the value of this data was recognized as these data provide cheap and easy access to valuable feedback about products, services, policies, and news. It further provides opportunities to leverage these data, process them automatically and extract knowledge and information from them. Such systems have applications in many different domains, such as political elections, business intelligence, decision support systems, government intelligence and medical decision-making and mental health. The importance of extracting and analyzing opinions from the Web was discussed in various research works, like socio-political studies Adamic and Glance (2005) and Kato et al. (2008).

Previously, opinion mining mainly relied on public surveys and polls. One drawback of such traditional opinion mining approaches is their cost. Moreover, they are not thoroughly up-to-date as there is often a gap between the emergence of an issue/event/news and conducting a survey. Recent studies such as Hill et al. (2013) showed that opinion mining from online social media is correlated to traditional approaches such as polls and surveys.

Our research is about the automatic opinion mining of social media data, as manually processing of sheer volume of such unstructured data is not feasible. This research area, known as opinion mining and sentiment analysis, is considered as a subfield of data mining, dealing with unstructured data. The growing interest to employ user-generated information on the Web to extract and determine individuals opinions has led to substantial attention to this research area. However, most of these efforts were towards subjectivity analysis of customer reviews of concrete entities such as products or movies (Pang et al., 2002; Dave et al., 2003; Pang and Lee, 2005).

For several reasons, prior works suggested for sentiment analysis in reviews has lower performance when applied on social media posts. One of the key reasons is the different nature of these two types of text genres. Reviews are typically highly focused where the authors often express their opinion directly about the product or movie, or their various aspects. In social media, users express their opinions about any topics and answer to each other posts in more

indirect ways, often involving sarcasm and irony.

People often express opinions towards various target entities in posts on social media such as online forums, blogs, Twitter, Youtube and Instagram. Social media platforms can be classified into social networks, blogs and their comments, microblogs, forums, social bookmarks, wikis, media sharing, and social news (Farzindar and Inkpen, 2015). Our main focus in this thesis is on microblogs (particularly tweets) and news comments.

Twitter has been considered as a corpus for studying public opinion as people express their positions in their tweets, directly or indirectly, about different events, products, issues, etc., because Twitter data is easy to collect. Analysis of Twitter data presents a unique set of challenges for technologies designed for more formal text, because the tweets are short, informal, and have special markers such as hashtags and emoticons. In this thesis, for the first time, we investigate the problem of stance detection on Twitter, whether the author is in favor of or against pre-chosen targets. These targets may or may not be referred to in a tweet, and they may or may not be the target of opinion in the tweet.

A huge quantity of news is published daily on the Web and several popular news websites like CNN.com and BBC.com allow their readers to express their opinion about the news by adding comments. These commentspheres can be considered as a special type of social media. Visualizing and summarizing the content of online news comments is of particular interest for decision makers who are seeking for public opinions and reactions to their news and events of interest. Most of the prior works for automatic processing of news comments have been focused on information retrieval tasks such as filtering, ranking, and summarization (Potthast et al., 2012). Nevertheless, opinion mining from online news comments has been less explored compared to other information retrieval tasks.

In this research, our goal is to determine netizens' attitude towards our targets of interest from their written social media texts. In (Perloff, 2010), attitude is defined as "a learned, global evaluation of an object (person, place, issue) that influences thoughts and actions", where attitudes are complex components of feelings and beliefs. While in sentiment analysis, the main focus is on affective aspects of attitude (Mohammad)), in this thesis, we emphasize more the cognitive dimension of attitudes and beliefs.

In online discussions, posts not only contain the position of the author towards the target but also convey the reasons behind the opinion or what action should be taken. This kind of subjectivity is called argumentation (Wilson and Wiebe, 2005). Argumentation analysis is more focused on the reasons for author's overall position. While mining social media for stance and sentiment is important, identifying arguments that people use to justify their views provides more insight about individuals or collectives (Schneider et al., 2012). Therefore an-

other objective of our research is to extract reasons and justifications behind authors' positions from their social media posts.

1.3 Problem Statement

In this thesis, our focus is on two main tasks: identifying stance and the reasons behind the stance, at the post level, from social media texts. In the rest of this section, we describe these tasks in more details (section 1.3.1 and 1.3.2).

1.3.1 Stance Detection

Stance classification task: The task we explore is formulated as follows: given a corpus of single posts in social media (tweet, online news comment, etc.) and a target entity (person, organization, movement, policy, etc.), automatic natural language systems should determine whether the author is in favor of the given target, against the given target, or whether neither inference is likely. Formally, given a set of social media posts D related to the target entity T, our goal is to determine the value of mapping $s_T : D \rightarrow \{favor, against, neither\}$ for any element $d \in D$. We propose to solve this task by learning the mapping using Machine Learning techniques. For example:

> Target: feminist movement
> Post: *Job should always go to best candidate, regardless of gender. Gender shouldn't even matter anymore, it's 2015! #PaulHenry*

The automatic system should classify this post as in favor of the target "feminist movement", by considering the text of the post alone, in absence of any other meta information like the author personal information, its interactions with other users, and thread structure.

Note that lack of evidence for 'favor' or 'against', does not imply that the author is necessary neutral towards the target. The post might be also classified as 'neither' if it is a mixture of opinions or does not have enough evidence to deduce the stance from the post text. The latter is a common phenomenon in our provided datasets in this thesis. For example:

> Target: atheism
> Post: *True community is being able to disagree and still love one another. #roadtolife*

From this single post, it is not possible to deduce the position of the author towards 'atheism'. We do not have 'neutral' label separately, as the number of social media posts from which we can infer neutral stance is expected to be very small.

Hence, stance detection, as we defined here, is a three-way text classification task. However, it is a difficult task compared to other text categorization problems, as stance can be expressed in various complicated ways in which, to identify the overall position, inference and background information might be required. For instance:

> Target: Bernie Sanders
> Post: *Be prepared - if we continue the policies of the liberal left, we will be #Greece*

From the following post, a human can deduce that the author is likely against 'Bernie Sanders'. The inference in a human mind is done by linking the "liberal left" to the 'Bernie Sanders'. It also requires background information about the Greek government debt crisis. In the next example, it can be inferred that the author supports 'legalization of abortion' from the written post. Here, the word *abortion* is not used and it is referred to as *control over body*.

> Target: legalization of abortion
> Post: *Today I am grateful to have the right to control my body without govt influence. #antichoice leaders want to stop this.*

One of the key challenges in stance classification is that the target of interest may not be the same as the target of the opinion. In all opinion mining applications, we are interested to collect opinions about our targets of interests, which can be concrete entities such as products or movies, or can be more abstract issues or ideologies. Often, opinions are expressed towards someone/something aligned with the target entity, from which we can infer the author's stance. As humans, we have the ability to infer the position towards the target of interest given opinions about other entities and the relationship between them in that context.

For instance, in electoral campaigns, it is extremely important for candidates to track public opinions about themselves. Thus, the target of interest is a particular candidate such as 'Donald Trump'. However, people may tweet about other candidates or parties from which their position towards 'Donald Trump' could be inferred. For example:

> Target: Donald Trump
> Post: *Jeb Bush is the only sane candidate for 2016.*

Here, the target of opinion is 'Jeb Bush'. However, we are interested in the position of the author towards 'Donald Trump'. It is deductible from the text that the author is against 'Donald Trump' since he was one of the other candidates for U.S. election of 2016.

Stance classification is related to sentiment analysis, but there are several differences between the two tasks. The key difference is that one may express favor (or disfavor) towards a target by using positive or negative language. For example:

Target: legalization of abortion
Post: *So not only are antichoice strongly against pregnant people's human rights, they're also homophobic. Shocker.*

We can infer that the author is likely to have favorable position towards 'legalization of abortion', while the post expresses a negative sentiment. Also note that the target of interest can be expressed in different ways, which impacts whether the instance is labeled 'favor' or 'against'. For example, if we rephrase the target here as 'pro-life movement', then the author is against the target. Thus, stance detection cannot simply be accomplished by mapping labels provided by a sentiment system, for example, positive labels to favor, and negative labels to against.

Another difference is that stance can be expressed without using sentiment-bearing words. For example, a written post might not be in support of or against anything, but it has some information from which we can infer the author's overall position towards the target. As an example:

Target: Mammography does not reduce breast cancer deaths
Post: *Studies show this over and over. The value of mammograms is vastly over-rated.*

Here, the target is a statement about the value of mammographies, and the author expressed her favorability towards it by referring to a fact that the truth of this statement has been shown by previous studies, while the language used in this post has neutral sentiment.

Stance classification can be considered as complementary to sentiment analysis because we often care about the author's evaluative outlook towards specific targets, rather than simple positive/negative classification. Nevertheless, for more focused texts like reviews of concrete entities, sentiment analysis techniques might be sufficient, as the text is often directly about the target entity and its aspects.

We finish this section by addressing different ways one can convey her stance. The simplest way to express position is to ***directly*** state the opinion towards the target; however, we often prefer to ***indirectly*** communicate our position by pointing to other persons/entities/ideas/issues that are related to the target. Furthermore, stance can be expressed by supporting or attacking arguments (facts or reasons) related to the target. For example:

Target: legalization of abortion
Post: *The woman has a voice the doctor has a voice. Who speaks for the baby? I'm just asking.*

We can infer from this post that the author is most likely against 'legalization of abortion' by bringing arguments in the form of a question, about the rights of unborn babies. Finally, rather

than explicitly expressing the stance, one can express her stance by framing a topic/target using biased terms and expressions.

In this thesis, we also explore the problem of detecting the stances expressed towards multiple targets in a single social media post. Prior work has often treated targets and the associated subjectivity independently, ignoring the potential (sometimes very strong) dependency that could exist among them. Here, we relieve the independence assumption and seek to jointly model the subjectivity expressed towards multiple related targets.

Multi-target Stance classification task: given a social media post and k related target entities, automatic natural language systems must jointly learn the overall position of the author towards the given targets where one prediction has a potential effect on the other ones. Formally, given a set of social media posts D related to the k target entities $T_1, ..., T_k$ our goal is to determine the value of mapping $s : T_1 \times ... \times T_k \times D \rightarrow \{favor, against, neither\}^k$ for each post $d \in D$. Our approach to solve this task is by learning the mapping using Machine Learning techniques. For example, consider the tweet and target pair:

> Targets: Donald Trump & Hillary Clinton
>
> Tweet: *Looking at the List of PC's for 2016 is like looking at the McDonalds Menu. You just know that shit is bad for you.*

The automatic system should determine from the above post that the author is most likely against both given targets. Note that the stances expressed towards two political candidates are not necessarily opposite of each other.

1.3.2 Reason Classification

As mentioned before, we are not only interested in detecting the position of the author towards a target of interest, but also in the reasons behind her position. Reason classification was first introduced as a separate task in Boltuzic and Šnajder (2014) in which the arguments were identified from a domain-dependent predefined list of arguments, and it has been called argument tagging. An argument tag is a controversial aspect in the domain that is abstracted by a representative phrase/sentence (Conrad et al., 2012). In this thesis, the terms argument tagging and reason classification refer to the same task and may be used interchangeably.

Reason classification task: given a single post in social media and a predefined list of possible reasons to back up the stance towards the target, label the post by one, more than one, or none of those argument tags.

There are several related works on analyzing argument structure in online interactions in discussions and forums (Ghosh et al., 2014; Sridhar et al., 2015; Abbott et al., 2011; Mukherjee and Liu, 2013). However, the focus of our research is not on interactions between different users and posts; we mainly aim at argument mining from a single post without considering the thread structure.

1.4 Contributions

In this section, we present the intended contributions of this thesis. These contributions are organized in three main categories: providing benchmark datasets, exploring and proposing new models and algorithms for stance detection and argument tagging, and fostering more research on stance detection by organizing a shared task on stance detection and analyzing the interaction between sentiment and stance.

- Providing benchmark datasets:
  - We created the first dataset of Twitter data labeled for both stance and sentiment. More than 4,500 tweets are annotated for whether one can deduce favorable or unfavorable stance towards one of six targets 'Atheism', 'Climate Change is a Real Concern', 'Feminist Movement', 'Hillary Clinton', 'Legalization of Abortion', and 'Donald Trump'. Each of these tweets is also annotated for whether the target of opinion expressed in the tweet is the same as the given target of interest. Finally, each tweet is annotated for whether it conveys the positive, negative, or neutral sentiment.
  - We created the first dataset of online news comments labeled for both stance and the reasons behind it. We manually annotated more than 1,000 posts collected from various news agency websites corresponding to the news that covered the controversial study published in British Medical Journal about breast cancer screening. Each post is also annotated for the intensity of the stance. in addition, the news comments are manually labeled by argument tags from a predefined list of tags (organized in a hierarchical tree structure).
  - We created the first multi-target dataset of tweets labeled for more than one target per post for stance. More than 4,400 tweets are annotated for whether one can deduce favorable or unfavorable stance towards two targets simultaneously. We

collected tweets related to 2016 US elections, with three target pairs for our Multi-Target Stance Dataset: 'Donald Trump and Hillary Clinton', 'Donald Trump and Ted Cruz', and 'Hillary Clinton and Bernie Sanders'. None of the previous datasets for stance annotated for more than one target per post.

- Exploring new models and algorithms:
  - We developed a state-of-the-art stance detection system by exploring several standard surface, semantic, and syntactic text classification features. We use a linear-kernel SVM classifier that relies on features drawn from the training instances—such as word and character n-grams—as well as those obtained using external resources—such as word-embedding features from additional unlabeled data.
  - We proposed a novel framework for argument tagging based on topic modeling. Unlike other machine learning approaches for argument tagging which often require a large set of labeled data, the proposed framework is minimally supervised and merely a one-to-one mapping between the pre-defined argument set and the extracted topics is required. The proposed framework has comparable results with the supervised framework on our news comments dataset.
  - We developed a stance detection system by leveraging automatically-extracted argument tags in addition to other standard text classification features. These additional features can capture the correlation between argument tags and stance labels. Adding extracted argument tags to linear-kernel SVM classifier with n-gram features improved classification performance from 62.90% to 64.55% in terms of average F-score.
  - We jointly modeled the stance expressed towards multiple targets by exploring deep neural networks. We adopted an attention-based encoder-decoder framework for this task and showed its effectiveness. It outperformed several alternative models that jointly learn dependent subjectivity through cascading classification or multi-task learning, as well as models that independently predict the subjectivity towards the individual targets.
- Fostering more research on stance detection:
  - For the first time, we organized a shared task competition on stance detection, the SemEval-2016, Task #6: Detecting Stance from Tweets. Two tasks were proposed. Task A is a traditional supervised classification task where 70% of the annotated

data for a target is used as a training and the rest for testing. For Task B, we use as test data all of the instances for a new target (not used in task A) and no training data is provided. Our shared task received submissions from 19 teams for Task A and from 9 teams for Task B.

- We conducted several experiments to better understand stance detection and its interaction with sentiment. We investigated the extent to which stance can be determined simply by traditional sentiment analysis (identifying positive and negative language). We used gold sentiment labels to determine stance and showed that it only helps determine stance to some extent, where the f-score is considerably lower than our stance classifier. Furthermore, ee applied the stance detection system as a common text classification framework, to determine both stance and sentiment. We showed that, while sentiment features are markedly useful for sentiment classification, they are not as useful for stance classification.
- We analyzed possible reasons for the lower performance of the stance classifier compared to sentiment classifier. We observed that stance detection is particularly challenging when the tweeter expresses an opinion about an entity other than the target of interest. (The text classification system performs close to majority baseline for such instances.) We concluded that, for these more difficult cases, more sophisticated approaches based on natural language inference might be required.

1.5 Outline

The rest of this thesis is organized as follows:

- In Chapter 2, several state-of-the-art works related to our research are reviewed. We start by presenting the main sentiment analysis and opinion mining approaches, and we continue by investigating previous works on stance classification and argument mining. Finally, as topic modeling and deep learning technologies are used in our proposed methods, we present these models in more details and explore their efficiency in various NLP tasks.

- Chapter 3 presents the dataset of Twitter data annotated for both stance towards given targets and sentiment. It also shows the efficiency of different semantic, syntactic, and surface features on stance classification and on sentiment classification. Later in this chapter, we explore two approaches to use additional unlabeled tweets in stance detection: distant supervision and word embeddings.

- In Chapter 4, we first describe our News Comments Dataset that we collected and manually annotated for stance and the arguments behind it. Then, a novel framework for joint modeling of arguments and stance is presented. This framework is mainly based on topic modeling and the efficiency of different topic modeling approaches is investigated in this framework.

- In Chapter 5, we discuss the task of joint prediction of stance towards multiple related targets. This chapter presents our Twitter dataset and proposed framework for multi-target stance detection. The proposed framework leverages neural models to capture the potentially complicated interaction between subjectivities expressed towards multiple targets.

- Chapter 6 concludes the thesis and proposes other lines of the research that can be explored in the future.

Figure 1.1 illustrates the key contributions of this research as organized in Chapters 3, 4 and 5. Chapters 3 and 5 are both focusing on stance detection in Twitter data, while in Chapter 4, stance in Online News Comments and the reasons behind it are explored. In Chapters 3 and 4, we address single target stance detection that is later extended to multi-target stance detection in Chapter 5.

1.6 Published Papers

- Parinaz Sobhani, Xiaodan Zhu, and Diana Inkpen. A dataset for multi-target stance detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, Valencia, Spain, April 2017

- Parinaz Sobhani, Saif Mohammad, and Svetlana Kiritchenko. Detecting stance in tweets and analyzing its interaction with sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 159–169, Berlin, Germany, August 2016

- Parinaz Sobhani, Diana Inkpen, and Stan Matwin. From argumentation mining to stance classification. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 67–77, Denver, CO, June 2015

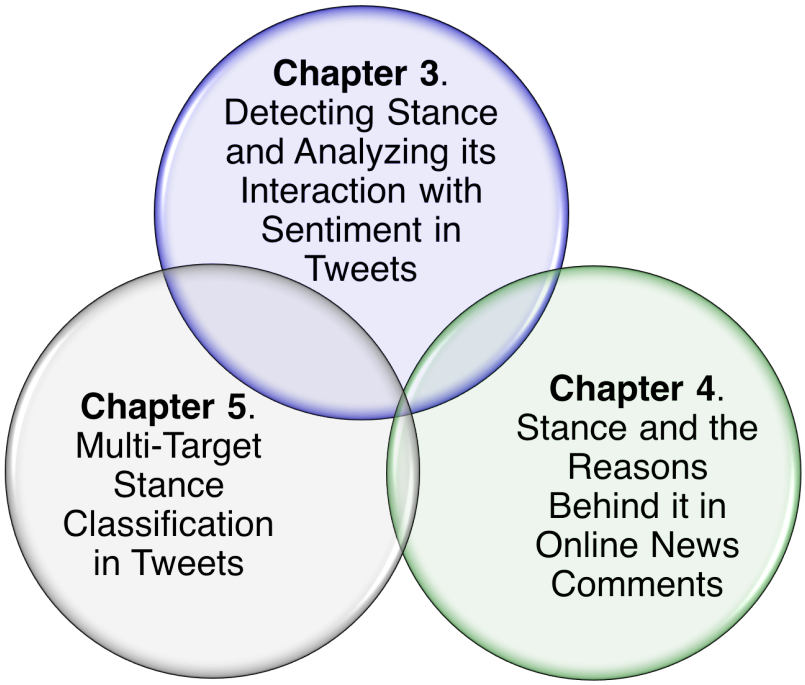

Figure 1.1: The key contributions of this research as organized in the Chapters 3, 4 and 5

- Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. Long short-term memory over recursive structures. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1604–1612, July 2015b

- Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. *Special Section of the ACM Transactions on Internet Technology on Argumentation in Social Media*, In Press, 2016d

- Parinaz Sobhani, Herna Viktor, and Stan Matwin. Learning from imbalanced data using ensemble methods and cluster-based undersampling. In *New Frontiers in Mining Complex Patterns - Third International Workshop, Held in Conjunction with ECML-PKDD, Revised Selected Papers*, pages 69–83, September 2014

- Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. Dag-structured long short-term memory for semantic compositionality. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 917–926, San Diego, California, June 2016

- Xiaodan Zhu, Hongyu Guo, and Parinaz Sobhani. Neural networks for integrating com-

positional and non-compositional sentiment in sentiment composition. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 1–9, Denver, Colorado, June 2015a

- Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. A dataset for detecting stance in tweets. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3945–3952, may 2016b

- Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California, June 2016a

Chapter 2

Background and Related Work

In this chapter, we review similar works to our research in the fields of machine learning and computational linguistics. Stance detection is closely related to sentiment analysis. Sentiment analysis tasks are formulated as determining whether a piece of text is positive, negative, or neutral, or determining from the text the speaker's opinion and the target of the opinion (the entity towards which opinion is expressed). In section 2.1, we further review computational methods and common text classification features for sentiment analysis in prior works. Later, in that section, we explore possible subtasks in the area of opinion mining and sentiment analysis that are closely-related to stance classification, such as aspect-based sentiment analysis.

We continue this chapter by reviewing recent works for stance classification (section 2.2) and available resources and datasets. As in this thesis, we additionally explore the problem of reason classification and argument tagging, in section 2.3, we review related works in the area of argumentation analysis.

In this research, several computational methods such as topic modeling, word embeddings, and deep neural networks are leveraged for stance and reason classification. To provide sufficient background for the next chapters, we explain these methods in detail in sections 2.4 and 2.5.

2.1 Sentiment Analysis and Opinion Mining

Recently, opinion mining and sentiment analysis have emerged as one of the most active areas in NLP in both academia and industry. Lately, more than 7,000 articles related to this field have been published (Feldman, 2013). Additionally, several companies have developed tools and packages for related applications. The main reason for this wave of popularity is the rapid

growth of social media such as Twitter, Facebook, news comments, and customer reviews. These media further provide cheap and easy access to huge subjective text and consequently, opportunities to leverage these data to extract knowledge and information in various domains, from political elections to financial services.

Originally, opinion mining referred to polarity detection, while sentiment analysis was mostly used for the extraction of affective information from text. However, nowadays these two terms are representing the same field and are used interchangeably (Liu et al., 2012). There are many challenges for automatic sentiment analysis and opinion mining that make this task more complicated than other text classification tasks. First, an opinion can be expressed in complex manners that cannot be identified by a single word or phrase. Another challenge is that opinion expressions are domain and context dependent.

Subjectivity detection is mainly focused on determining the presence or absence of sentiment in the given sentence or phrase. Wiebe et al. (1999) defined *subjective text spans* as those that "are used to communicate the speaker's evaluations, opinions and speculations" This is largely close to classifying text as opinionated vs. non-opinionated. Subjectivity information has been shown to help in improving the accuracy of sentiment classifiers (Wilson et al., 2005c).

Each opinion has three major dimensions: target, orientation or polarity, and strength or intensity. The opinion target is the entity that the opinion is about. The term "entity" can denote a product, an event, a person, an idea, etc. The opinion polarity can be positive, negative or neutral. In order to measure opinion strength, a number in a predefined scale can be assigned to the text. Based on these dimensions, several tasks are defined as subtasks of opinion mining including subjectivity detection, polarity classification and target identification. Most of the current approaches focus only on polarity classification.

For certain applications, it is crucial to identify the opinion holder. For instance, a news article may contain several opinions from different persons and it is necessary to distinguish between opinions of different opinion holders. Other applications are aggregating all opinions of an opinion holder and grouping people based on their opinions.

Another goal of opinion mining is to detect opinions toward a particular entity. Often, this requires identifying the target of the opinion and its relationship with the target of interest. To identify opinion targets, available techniques for Named Entity Recognition (NER) can be exploited (Hobbs and Riloff, 2010; Jakob and Gurevych, 2010); however, this problem is more complex because the same entity can be referred to via different words and phrases.

In (Liu, 2012), it has argued that an opinion has two more dimensions, which are the time when the opinion was expressed and the aspect of the opinion target entity. Without

considering the time dimension of the subjectivity, opinion tracking in a given period of interest is not possible. Regarding these added dimensions, other possible related tasks are opinion aspect extraction and monitoring opinion over time (Zhao et al., 2010b; Brody and Elhadad, 2010; Ku et al., 2006).

In the thesis, we are mainly focusing on opinion mining in ideological and political domains. However, the majority of the studies on opinion mining have mainly focused on reviews of products or services as they are easier to process compared to other text genres such as ideological or political discourse. Mullen and Malouf (2006) is one of the early works that investigated the challenges of opinion mining in political domains. They noted that "political attitudes generally encompass a variety of favorability judgments toward many different entities and issues. These favorability judgments often interact in unexpected or counterintuitive ways."

Opinion mining can be applied on different levels: term-level (Wilson et al., 2013b), phrase-level (Wilson et al., 2005c), sentence-level (Li et al., 2010) or document-level (Pang et al., 2002). Opinion mining at each of these levels has its own application and challenges. The major challenge of sentiment analysis at document-level is that a document can be about different entities or different aspect of a single entity. Moreover, it may contain multiple opinions about an entity. On the other hand, the main issue with opinion mining at finer granularities is the lack of context. Often, the meaning of a piece of text is composed through several clauses and sentences and by analyzing them separately, part of the meaning might be lost. In chapters 3 and 5, we explore Twitter opinion mining. Each tweet can be considered as a sentence, which means that sentiment analysis is applied at sentence-level. Later in the thesis, in chapter 4, stance detection in news comments is investigated. News comments are often longer and opinion mining is mainly applied at document-level.

Liu (2010) classified opinions as direct and indirect. Direct opinions are easier to analyze, while indirect opinions are expressed in more complicated ways. For instance, an opinion about an entity may be expressed "based on its effects on some other entities" (Liu, 2010). Similarly, opinions can be grouped as explicit and implicit. An explicit opinion is a subjective statement, while an implicit one is an objective statement that entails an opinion without using any opinion expressions (Liu, 2010). In chapter 3, we show that the main challenge in automatic Twitter stance detection is indirect opinions.

2.1.1 Computational Approaches to Sentiment Analysis

Prior works for opinion mining and sentiment analysis can be divided into two major groups, lexicon-based approaches and statistical approaches. Lexicon-based approaches mainly rely on the presence of words or phrases with clear affect in order to assign polarity labels (Ding et al., 2008). Statistical methods attempt to predict labels based on several semantic or syntactical features extracted from the text (Pang et al., 2002). Additionally, there are hybrid approaches that use lexicons as extra features (Kiritchenko et al., 2014b).

Lexicon-based Approaches

The majority of lexicon-based approaches work at sentence-level or finer-level granularities. Typically, they follow the four steps suggested by Ding et al. (2008): extracting affective terms or phrases based on one or several lexicons, detecting sentiment shifters such as negations that reverse the sentiment, identifying contrary text spans like *but*-clauses, and finally aggregating the sentiment scores calculated in the previous steps.

One of the main limitations of lexicon-based approaches is that opinions are not only expressed with affective terms/phrases, but there are numerous other expressions or terms that may imply opinion and most of them are domain and context-specific. In (Liu, 2010), a set of complex rules and patterns suggested in order to cover all possibilities of opinion occurrences, but they could not recognize all opinion expressions. Finally, they concluded that sixty percent of opinions are expressed by affective words and the rest of opinions are expressed in considerably different and domain-specific manners (Liu, 2010).

One major drawback of lexicon-based methods is that they build a global lexicon, while several opinion bearing words are domain specific and may have different polarities in different domains. To overcome this drawback, corpus-based approaches were proposed. The primary purpose of these approaches is to extend the existing list of affective words with domain-specific ones, using the target corpus. These methods are mostly based on the rule of cohesion that assumes words within a sentence and in neighboring sentences have the same polarity, unless a sentiment reverser term such as “but” appears in the middle (Kaji and Kitsuregawa, 2007; Kanayama and Nasukawa, 2006).

The main problem with corpus-based approaches is that even within a corpus, words may have different sentiment based on their context. As a remedy, Wilson et al. (2005c) investigated subjectivity and sentiment at phrase or expression level. Similarly, in Yessenalina and Cardie (2011), the context of sentiment words was represented by word vectors. Another challenge is that factual words and expressions may be opinionated and they cannot be captured by lexicon-

based methods.

How to build a polarity lexicon Since manually building a lexicon is time-intensive and requires human experts, most of the lexicons are built automatically or with minimal supervision (the supervision is used only to control the quality of the extracted lexicon). One way to automatically build a lexicon is to exploit available resources/dictionaries such as WordNet. The general methodology is to use a bootstrapping approach that starts with a small number of seed words and extend them based on the synonym and antonym relations in the dictionaries (Hu and Liu, 2004; Blair-Goldensohn et al., 2008). Similarly, a score can be assigned to each word based on its distance to reference (seed) words in the WordNet semantic graph (Kamps et al., 2004).

In (Mohammad et al., 2013a), another approach for automatically building lexicons was suggested. The method exploited Twitter hashtags. Hashtags can be regarded as conveying freely available (albeit noisy) human annotation of sentiment. More specifically, certain words in tweets are specially marked with the hash character (#) to indicate a topic, a sentiment polarity, or an emotion such as joy, sadness, anger, or surprise. With enough data, such artificial annotations can be used to learn the sentiment of a single word or sequences of n words (n-gram) by their likelihood of co-occurring with hashtagged words. In the mentioned work, a collection of seed hashtags closely related to *positive* and *negative* concepts were used together with a large set of tweets that contain at least one positive or negative hashtag, in order to build an annotated corpus. Subsequently, the association score for an n-gram w was calculated from these pseudo-labeled tweets as follows:

$$score(w) = PMI(w, positive) - PMI(w, negative) \tag{2.1}$$

where PMI stands for pointwise mutual information and the two terms in the formula calculate the PMI between the target n-gram and the pseudo-labeled positive tweets, as well as that between the n-gram and the negative tweets, respectively.

Machine Learning Approaches

Sentiment classification can be considered as one of the applications of text categorization. Hence, supervised learning algorithms such as Support Vector Machines (SVM), Naive Bayes (NB), Logistic Regression, Decision Trees and rule learning algorithms can be employed. The main drawback of these statistical methods is that, in order to train a model, a dataset with annotated instances (training set) is required. Mostly, a training set needs to be manually

annotated, and this can be tedious and time-consuming particularly for large datasets.

SVM is one of the state-of-the-art learning algorithms that has proven to be effective on text categorization tasks and robust on large feature spaces. It has been used in many works on sentiment classification (Pang et al., 2002; Mullen and Collier, 2004; Pang and Lee, 2005; Barbosa and Feng, 2010). One of the earliest works is Pang et al. (2002), where the performance of SVM is compared to NB and Maximum Entropy (MaxEnt) classifiers for movie reviews sentiment classification. In their experiments, SVM outperformed the other two classification algorithms.

In (Wilson et al., 2009), other classification algorithms such as a rule learning algorithm named Ripper, a memory-based learning k-nearest neighbor (KNN), and the boosting classifier AdaBoost were used for polarity classification. Recently, other advanced classification methods like Deep Neural Networks have shown promising results for sentiment classification (Moraes et al., 2013; Socher et al., 2013; Zhu et al., 2016).

Sentiment analysis may be also achieved by unsupervised approaches. One of the first methods for unsupervised sentiment classification is Turney (2002), in which terms that follow predefined syntactic patterns are extracted. These patterns are based on part-of-speech (POS) tags of the extracted terms. In another unsupervised approach, Taboada et al. (2011) used a dictionary of terms consisting of their orientation and strength. Moreover, negation and intensifiers are taken into consideration.

A key factor in the performance of any statistical system for sentiment detection is feature engineering: the set of features that are selected to represent the text. In the next section, we review most popular features such as n-grams, syntactical and lexicon-based features for sentiment analysis.

2.1.2 Feature Engineering for Sentiment Classification

In order to automatically label a piece of text as positive, negative, or neutral using machine learning algorithms such as SVM and MaxEnt, an appropriate numeric representation for text should be obtained. Similar to other text classification tasks, in sentiment analysis, a great effort has been devoted to identify the best set of features for this particular task. This process is called feature engineering and has a large impact on the performance of a classifier.

A text can be represented by several different features. Vectors of absence or presence of a word or sequence of words (word n-grams) or their frequencies are widely used to represent text (Pang et al., 2002) for sentiment analysis. Similarly, character n-grams showed to be effective in sentiment detection (Kiritchenko et al., 2014b). Previous works made use of

different weighting schemes to give weights to these n-grams based on their importance for the classifier (Martineau and Finin, 2009; Paltoglou and Thelwall, 2010). One of the most popular schemes is Term Frequency-Inverse Document Frequency (TF-IDF) in which a term's weight increases as it is more frequent in a target document, but decreases as it appears in more documents of the corpus.

Even though it is a common practice in text classification to represent a text by its character or word n-grams, this representation has several limitations. The most important ones are: ignoring the word order, discarding the syntactic structure and semantic dependencies, and disregarding the scope of negations. To fix some of these problems, we can increase n in terms of the length of the sequence of words or characters to capture the word order in short context, but this increases the sparsity and has a negative effect on the performance of the classifier. The other key problem with this representations is that, in the feature space, all terms are equally distant, while they might be semantically close.

There has been growing research on extracting other features for sentiment detection. As the accuracy of automatic semantic and syntactic parsers improved, deeper features based on semantic and syntactic structures of the text were explored. The other source to consider for feature extraction are common-sense knowledge bases.

Syntactic-based features: The efficiency of syntactic features for sentiment prediction has been widely investigated in different studies (Mullen and Collier, 2004; Xia and Zong, 2010, 2011)). Syntactic information can be incorporated to handle the problem of polarity shift (Kennedy and Inkpen, 2006; Na et al., 2004; Ng et al., 2006). Polarity shift is the language phenomenon that reverses the sentiment of a text. In (Kennedy and Inkpen, 2006), three types of polarity shifters, namely negations, intensifiers, and diminishers were considered. Negation reverses the polarity, whereas intensifiers and diminishers alter the sentiment strength. In their work, syntactic parsing is proposed to identify these three types of shifters. In another similar work, negation was modeled by considering POS patterns (Na et al., 2004). In (Gamon, 2004), a diverse set of linguistic features mainly based POS information was extracted and consistent improvements in the accuracy of sentiment classifier were reported. Similarly, Wilson et al. (2005c) extracted a large set of features from dependency parse tree to capture negation for recognizing the contextual polarity of words and phrases.

Lexicon-based features: Incorporating external knowledge like lexicons into statistical models is usually beneficial, considering that most datasets for sentiment analysis have a relatively limited amount of training data, and external prior knowledge could help to cover missing semantics. These external knowledge sources have been incorporated in statistical learners using various approaches (Kiritchenko et al., 2014a; Zhu et al., 2016). One of the most pop-

ular approaches is to calculate the frequency of positive or negative terms or similarly, the summation of the scores of affective terms.

There are two main types of resources for the prior sentiment of affective terms: (1) automatically learned lexicons from an external, large corpus, and (2) manual lexicons where the sentiment of n-grams assigned by human annotators. Manual lexicons include a widely-used sentiment lexicon, such as the MPQA Subjectivity Lexicon Wilson et al. (2005b), which encodes the prior knowledge that the human annotators have about the sentiment of words. MPQA, which draws from the General Inquirer (Stone et al., 1962) and other sources, has sentiment labels for about 8,000 words. The contained words marked with their prior polarity (positive or negative) and a discrete strength of evaluative intensity (strong or weak). The other popular lexicon is the NRC Emotion Lexicon (Mohammad and Turney, 2010). This lexicon has 14,182 words with their associations with eight emotion classes (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and two sentiment classes (negative and positive).

The major limitation of manual lexicons is their low coverage as they have a limited number of terms and the fact that manually creating larger lexicon is costly. Several approaches for automatically or semi-automatically building of lexicons were suggested (Turney and Littman, 2003; Kiritchenko et al., 2014b), as described earlier in section 2.1.1. For example, the SentiWordNet lexicon (Baccianella et al.) is created with a combination of supervised learning and manual annotation.

2.1.3 Sentiment Analysis for Social Media Data

Opinion mining for online news comments: Most of the prior works for automatic processing of news comments have been focused on information retrieval tasks such as filtering, ranking, and summarization (Potthast et al., 2012). Opinion mining from news comments has been less explored compared to other information retrieval tasks.

Different approaches towards sentiment analysis of news comments, mainly polarity and emotion detection, have been elaborated. Most of the prior works used supervised learning algorithms. In (Zhou et al., 2010), different feature sets for sentiment analysis of news comments were compared. Chardon et al. (2013) explored the effect of using discourse structure for predicting news reactions. In (Zhang et al., 2012), a meta-classifier for tagging emotions (such as sadness, surprise, and anger) towards the news was proposed. In their method, the authors used two heterogeneous information sources: content-based information and emotion tags in the comments. Jakic (2011) suggested an approach for automatic prediction of the sentiment polarity in reactions to the news. In this work, the authors used domain-knowledge

transfer from a classifier trained on Twitter data.

Moreo et al. (2012) suggested a lexicon-based approach which can adapt to different domains. In their work, they designed and built a structured lexicon using WordNet relations. Zhao et al. (2010a) proposed an unsupervised algorithm for online comments classification by applying affinity propagation clustering on comments. They retrieved key sentences of each comment and subsequently extracted their Named Entities as the targets of comment. In another similar unsupervised approach for polarity classification of online news comments, Sun et al. (2011) used polarized word counts weighted by their distance to comments targets.

Twitter Sentiment Analysis: Twitter has been considered as a corpus of studying public opinion, as increasingly many people express their opinions implicitly or explicitly in their tweets. There is a vast amount of work in sentiment analysis of tweets, and we refer the reader to surveys (Pang and Lee, 2008; Liu and Zhang, 2012; Mohammad) and proceedings of recent shared task competitions (Wilson et al., 2013a; Rosenthal et al., 2015).

Twitter sentiment analysis provided unique challenges for methods mainly aimed to automatically process more formal text. Twitter posts are short and informal. There are plenty of special markers such as hashtags and emoticons, slang terms and inconsistent capitalization in tweets. Another issue with Twitter data is that they tend to not follow grammatical rules and typically, there are many misspelling and shortened forms of words. Consequently, new features for sentiment detection of Twitter data proposed in previous researches. In (Kiritchenko et al., 2014b), a variety of surface-form features such as presence/absence of positive and negative emoticons, hashtags, characters in upper case and elongated words (e.g., *sweeettt*) tailored for short, informal tweets.

In recent years, there has been considerable interest in analyzing political tweets for sentiment, emotion, and purpose in electoral tweets (Mohammad et al., 2015), determining political alignment of tweeters (Golbeck and Hansen, 2011; Conover et al., 2011a), identifying contentious issues and political opinions (Maynard and Funk, 2011), detecting the amount of polarization in the electorate (Conover et al., 2011b), and even predicting the voting intentions or outcome of elections (Tumasjan et al., 2010; Bermingham and Smeaton, 2011; Lampos et al., 2013). One of the more recent works, Lampos et al. (2013) analyzes tweets from UK and Austria and successfully predicts voting intention in more than 300 polls across the two countries.

2.1.4 Related Subtasks in Sentiment Analysis

There are several subtasks in opinion mining closely related to stance classification, such as biased language detection (Recasens et al., 2013; Yano et al., 2010), perspective identification (Lin et al., 2006) and user classification based on their views (Kato et al., 2008). Perspective identification was defined as the subjective evaluation of points of view (Lin et al., 2006). Greene and Resnik (2009) identified implicit sentiment or perspective and went beyond lexical indicators to show the importance of "syntactic packaging" in human sentiment judgments. To detect biased language, Recasens et al. (2013) extracted linguistic cues such as factive verbs, hedges and subjective intensifiers as features for their logistic regression classifier. Iyyer et al. (2014) used Recursive Neural Networks (RNN) to determine political bias in a text at sentence-level. In another work, users in online forums were classified based on their political orientation (Kato et al., 2008) where their approach was a combination of Point-Wise Mutual Information and Information Retrieval, Naive Bayes, and graph-based methods.

Deng and Wiebe (2014) investigated the relationships and interactions among entities and events explicitly mentioned in the text with the goal of improving sentiment classification. In stance classification, however, the predetermined target of interest may not be mentioned in the text, or may not be the target of the opinion in the text.

Another related research to stance classification is "target-dependent" sentiment classification (Jiang et al., 2011), in which, given a target query, a document is classified as positive, negative, or neutral, if it contains positive, negative, or neutral sentiment towards the query. The authors proposed an approach solving the problem of assigning irrelevant sentiment to tweets about a given target. By irrelevant sentiment, they mean those sentiments that are not towards the given target. In their approach, they incorporated a set of target-dependent features in addition to target-independent features similar to other sentiment classifiers. Target-dependent features included a set of relations between the extended target and verb, adjective and adverb phrases extracted from the dependency parse tree of the sentence. The extended target is represented by the target and all noun phrases related to it.

In aspect-based sentiment analysis (ABSA), the goal is to determine the sentiment towards aspects of a product such as the speed of a processor or the screen resolution of a cell phone. This area has received considerable attention over the last few years—there were two shared task competitions exploring this task in 2014 and 2015 (Pontiki et al., 2015, 2014). We refer the reader to SemEval proceedings for more comprehensive related work on ABSA (Pontiki et al., 2015, 2014). Mohammad et al. (2013b) and Kiritchenko et al. (2014a) came first in the 2013 Sentiment in Twitter and 2014 SemEval ABSA shared tasks. We use similar features to

the ones they used, in our stance classification framework in chapter 3.

2.2 Stance Classification

Over the past decade, there has been active research in modeling overall positions in user-generated contexts. However, the majority of the works focused on congressional debates (Thomas et al., 2006) or debates in online forums (Somasundaran and Wiebe, 2009b; Anand et al., 2011; Walker et al., 2012b; Hasan and Ng, 2014; Sridhar et al., 2014; Murakami and Raymond, 2010). The advantage of using such domains is that gold labels are given by the authors. Nevertheless, stance detection in other forms of user-generated contents like Twitter data and news comments are mostly unexplored.

Supervised learning has been used in almost all of the current approaches for stance classification, in which a large set of data has been collected and annotated in order to be used as training data for classifiers. In (Somasundaran and Wiebe, 2010), a lexicon for detecting argument trigger expressions was created and subsequently leveraged to identify arguments. These extracted arguments, together with sentiment expressions and their targets, were employed in a supervised learner as features for stance classification.

In (Anand et al., 2011), several features were deployed in a rule-based classifier, such as unigrams, bigrams, punctuation marks, syntactic dependencies and the dialogic structure of the posts. The authors showed that there is no significant difference in performance between systems that use only word unigrams and systems that also use other features such as Linguistics Inquiry Word Counts (LIWC) and POS generalized dependencies. The dialogic relations of agreement and disagreements between posts were also exploited by Walker et al. (2012b). In this research, we decided to focus only on the text of the user-generated contents as these relationships are not provided for our stance datasets.

Faulkner (2014) investigated the problem of detecting document-level stance in student essays by making use of two sets of features that are supposed to represent stance-taking language. Table 2.1 shows the summary of the features that have been used in various stance analysis studies. These features are divided into six main groups: n-grams, length-based, syntactic, sentiment, argumentative, and non-linguistic constraints.

Different machine learning algorithms are employed for automatic classification of overall position from unstructured text. While SVM and logistic regression were widely used in various studies (Walker et al., 2012a; Somasundaran and Wiebe, 2010), Conditional Random Fields (CRF) and Linear Integer Programming were employed in (Wang and Cardie, 2014) and (Hasan and Ng, 2013) to additionally capture agreement and disagreement in user interactions.

Features Type	**Features**	**Reference**
n-grams	Words unigrams and bigrams	Somasundaran and Wiebe (2010); Anand et al. (2011)
Length-based	Number of sentences, words, and characters	Sridhar et al. (2014)
Syntactic	Dependencies and generalized dependencies with respect to POS tags	Joshi and Penstein-Rosé (2009); Somasundaran and Wiebe (2009b); Anand et al. (2011)
Sentiment	Linguistics Inquiry Word Counts (LIWC), MPQA to select the subset of generalized dependency features and replaced opinion words with their sentiment	Lin et al. (2006); Anand et al. (2011); Somasundaran and Wiebe (2010)
Argumentative	Repeated punctuation, Initial unigram, bigram and trigram, Arguing Lexicon, Modal verbs	Anand et al. (2011); Somasundaran and Wiebe (2010)
Non-Linguistic constraints	Author constraints, User-interaction constraints, Ideology constraints	Lu et al. (2012); Walker et al. (2012b); Hasan and Ng (2013)

Table 2.1: List of features elaborated in different statistical learners for stance classification

The assumption in (Hasan and Ng, 2013) is that consecutive posts are not independent of each other and constraints on adjacent posts make the problem a sequence labeling task. In (Ahmed and Xing, 2010), an extension to the Latent Dirichlet Allocation (LDA) algorithm is used to model each word as the interaction of ideological and topical dimensions.

In one of a few works in stance detection in Twitter, Rajadesingan and Liu (2014) determined stance at user-level based on the assumption that if several users retweet one pair of tweets about a controversial topic, it is likely that they support the same side of a debate. In this research, we focus on detecting stance, as much as possible, from a single tweet. Features that help to this end will likely also be useful when there is access to multiple tweets from the same tweeter. In another work for Twitter stance detection, bi-directional Long Short Term Memory was used to encode the target and the tweet (Augenstein et al., 2016a). In that method, the representation of the tweet and the target depend on one another and the experiments demonstrated improvement over independently encoding the tweet and the target.

Textual Entailment Stance detection is related to textual entailment. In textual entailment, the goal is to infer a textual statement (hypothesis) from a given source text (Dagan and Glickman, 2004). Textual entailment is a core NLP building block and has applications in question answering, machine translation, information retrieval and other tasks. It has received a lot of attention in the past decade, and we refer the reader to surveys Androutsopoulos and Malakasiotis (2010) and Dagan et al. (2013) and proceedings of recent challenges on recognizing textual entailment Bentivogli et al. (2011), Marelli et al. (2014) and Dzikovska et al. (2016).

The task we explore in this research, stance detection in user generated contents, can be viewed as another application of textual entailment, where the goal is to infer a person's opinion towards a given target based on a single tweet written by this person. In this special case of textual entailment, the hypotheses are always fixed (the person is either in favor of or against the target). Furthermore, we need to derive not only the meaning of the tweet but also the attitude of the text's author.

2.2.1 Available Datasets

Online debates are a rich source of opinions and feedback about different social or political issues. Existing datasets for stance detection were mostly created from online debate forums like 4forums.com and createdebates.com (Somasundaran and Wiebe, 2010; Walker et al., 2012c; Hasan and Ng, 2013). The majority of these debates are two-sided and the data labels are often provided by the authors of the posts. The topics of these debates are mostly related to ideologically controversial issues such as gay rights and abortion.

Among all, the corpus of Somasundaran and Wiebe (2010) was one of the first stance dataset collected from different debating websites. Originally, data were collected for six topics: 'Existence of God', 'Health-care','Gun Rights','Gay Rights','Abortion' and 'Creationism'. Another available corpus is that of Hasan and Ng (2013) which was collected from *createdebates.com* for four topics: 'Abortion', 'Obama', 'Marijuana' and 'Gay Rights'. Similarly, posts are labeled with the labels provided by authors of the posts. This dataset is also annotated for reason labels. Two annotators were asked to tag each sentence by a reason from a given list of reasons. One issue with these two corpora is that they were automatically labeled and there might be irrelevant posts that are wrongly labeled as "pro" or "con".

Another popular corpus for stance classification is the Internet Argument Corpus (Walker et al., 2012c). This corpus was extracted from debates in *4forums.com* website where users can express their views and interact with others. This corpus is one of the biggest corpora for deliberation and debates about a broad range of topics such as 'Death Penalty' and 'Gay Marriage'. It has more than 11,800 discussions and 390,704 posts. From them, 6,144 posts about 10 topics were selected to be annotated for stance using Amazon's Mechanical Turk. Annotators were asked to label each post as "pro", "con" or "other".

In this thesis, we created the first dataset of tweets labeled for both stance and sentiment (see Chapter 3). Furthermore, we created the first dataset for stance and reasons behind it in online news comments (see Chapter 4). We also created the first multi-target stance dataset of Twitter data (see Chapter 5). None of the prior works created a dataset annotated for more than one target simultaneously.

2.2.2 SemEval-2016 Task 6: Detecting Stance in Tweets

Stance detection was one of the tasks in the SemEval-2016 shared task competition (Mohammad et al., 2016c), where for the first time we presented a shared task on detecting stance from tweets. Two tasks were proposed. Task A is a traditional supervised classification task where 70% of the annotated data for a target is used as a training and the rest for testing. For Task B, we use as test data all of the instances for a new target (not used in task A) and no training data is provided. Our shared task received submissions from 19 teams for Task A and from 9 teams for Task B. The highest classification F-score obtained was 67.82 for Task A and 56.28 for Task B. However, systems found it markedly more difficult to infer stance towards the target of interest from tweets that express opinion indirectly (towards another related entity).

Out of 19 participant teams for task A, most used standard text classification features such as n-grams and word embedding vectors, as well as standard sentiment analysis features such

as those drawn from sentiment lexicons (Elfardy and Diab, 2016; Krejzl and Steinberger, 2016; Patra et al., 2016; Zhang and Lan, 2016; Tutek et al., 2016). While, others used deep neural models such as autoencoders (Augenstein et al., 2016b), recursive (Zarrella and Marsh, 2016) and convolutional neural networks (Wei et al., 2016; Yuki et al., 2016; Vijayaraghavan et al., 2016).

Some teams polled Twitter for stance-bearing hashtags, creating additional noisy stance data Zarrella and Marsh (2016); Misra et al. (2016). Most of the teams elaborated continuous word representations in their models (Liu et al., 2016; Bøhler et al., 2016). These word vectors were derived from extremely large sources such as Google News, directly from Twitter corpora, or as a by-product of training a neural network classifier. Nine out of the 19 teams used some form of word embeddings, including the top three winning systems for task A (Zarrella and Marsh, 2016; Wei et al., 2016; Tutek et al., 2016). Seven of the 19 submissions made extensive use of publicly-available sentiment and emotion lexicons such as the NRC Emotion Lexicon (Mohammad and Turney, 2010), Hu and Liu's Lexicon (Hu and Liu, 2004), the MPQA Subjectivity Lexicon (Wilson et al., 2005b), and the NRC Hashtag Lexicons (Kiritchenko et al., 2014b).

The best result obtained for task A was that of Zarrella and Marsh (2016) with an overall F_{avg} of 67.82. Their approach employed two recurrent neural networks (RNN) classifiers: the first was trained to predict task-relevant hashtags on a very large unlabeled Twitter corpus. This network was used to initialize a second RNN classifier, which was trained with the provided Task A data.

The Task B teams varied wildly in terms of approaches to this problem. The top three teams all took the approach of producing noisy labels, with Wei et al. (2016) using keyword rules, *LitisMind* team using hashtag rules on external data, and Dias and Becker (2016) using a combination of rules and third-party sentiment classifiers. However, other teams attempted to generalize the supervised data from Task A in interesting ways, either using rules or multi-stage classifiers to bridge the target gap.

2.3 Argumentation Mining

In (Somasundaran et al., 2007), two types of opinions are considered: sentiment and arguments. While sentiment mainly includes emotions, evaluations, feelings, and stances, arguments are focused on convictions and persuasion. Even though mining social media for opinion and sentiment is important, identifying arguments that people may use to justify their views can provide more insight about individuals or collectives (Schneider et al., 2012).

Argument mining has been used in various domains including scientific papers (Teufel et al., 2009), legal documents (Hachey and Grover, 2005), or even product reviews (Albert et al., 2011); however, argument mining from social media data is more challenging compared to more formal types of texts such as scientific papers or news. One reason for this is the informal language of social media text where there are many misspellings, abbreviations, and slang. Other reasons are the use of indirect language like sarcasm and irony, lack of argument indicators, and implicit opinions (Schneider et al., 2012).

One of the most widely used argumentation schemes in the literature is the Claim-Premises scheme (Habernal et al., 2014) where an argument is defined as a single claim and a set of premises that either attack or support the claim (Besnard and Hunter, 2008). In this model, a claim is described as "the conclusion we seek to establish by our arguments" and premises are "connected series of sentences, statements, or propositions that are intended to give reasons of some kind for the claim" (Freeley and Steinberg, 2013). Although there are several other models for argumentation structure, the key advantage of this scheme is its simplicity. For instance, Toulmin (2003) suggested a model that allows distinction between different types of premises based on their role. The applicability of such more complex models on poorly-structured user-generated content, where some of the argument components are left implicit, is hardly feasible (Habernal et al., 2014).

In (Albert et al., 2011), argument mining for customer reviews was introduced in order to extract the reasons for positive or negative opinions and with the goal of information mining. In a different domain, Chenlo et al. (2014) proposed a framework that aims to help participants in online discussions. In their framework, textual entailment is employed on the Debatepedia dataset to label the relation between each pair of arguments as entailment or contradiction. There are several other works on analyzing argument structure in online interactions such as discussions and forums (Ghosh et al., 2014; Sridhar et al., 2015; Abbott et al., 2011; Mukherjee and Liu, 2013). However, the focus of our research is not on the interaction between different users and posts. We aim at argument mining for a single post, without considering the thread structure.

Argument tagging was first introduced as a separate task in Boltuzic and Šnajder (2014) in which the arguments were identified from a domain-dependent predefined list of arguments. An argument tag is a controversial aspect in the domain that is abstracted by a representative phrase/sentence (Conrad et al., 2012). Argument tagging can be applied at different text granularities. In (Conrad et al., 2012), a model for argument detection and tagging at sentence-level was proposed. In (Hasan and Ng, 2014), a reason classifier for online ideological debates is suggested. In that method, document-level reason classification is leveraged by aggregating all

sentence-level reasons of a post.

For most of the existing argument tagging approaches, the list of possible arguments needs to be identified manually. Recently, in (Swanson et al., 2015), a new approach for automatically extracting arguments was proposed. Their method is similar to sentence extraction in multi-document summarization in which sentences that clearly represent an argument facet are identified. Their work is based on the hypothesis that the best candidates for argument tags are marked by cues such as high semantic density and certain discourse relations.

2.4 Topic Modeling for Text Classification and Clustering

Topic models are a group of methods in NLP that attempt to extract hidden topics from a corpus of documents. Previously, it was assumed that each document has one associated topic, but this assumption is relaxed in topic modeling by considering each document as a mixture of different topics. These models were used for different text genres such as scientific journals (Wallach, 2006), news (Newman et al., 2006), blogs and tweets (Mei et al., 2007; Resnik et al., 2015). However, topic modeling in more informal documents is more challenging due to the less organized and unedited style of these documents.

One of the first algorithms that have been used for topic modeling is Latent Semantic Analysis (LSA) (Deerwester et al., 1990). LSA was originally designed for automatic document indexing and information retrieval in order to provide a low-dimensional representation for a document. The key advantage of this representation is its ability to capture document similarities based on their relevance in semantic space. Later, this model was extended as Probabilistic Latent Semantic Indexing (PLSI) (Hofmann, 1999) to deal with words with different meanings and usages.

Non-Negative Matrix Factorization (NMF) (Lee and Seung, 2001) has been also extensively used for text clustering and topic modeling (Xu et al., 2003; Shahnaz et al., 2006). In this model, a given matrix $V \in R^{n \times m}$ is factorized into two matrices $W \in R^{n \times r}$ and $H \in R^{r \times m}$ with the constraint that none of these matrices can have negative elements (the non-negativity property makes these matrices easier to analyze):

$$V \approx WH \tag{2.2}$$

Here, W corresponds to the term-topic matrix, while H is the topic-document matrix. The advantage of NMF is that it can generate lower dimension factors (r is significantly smaller than m and n). Assuming that each document is generated from a small set of hidden topics, H

can capture these latent topics. One of the most popular methods to solve the above equation is by alternative non-negative least squares using projected gradients (Lin, 2007). Another advantage of NMF is that it allows correlations between latent topics (Arora et al., 2012).

One of the most popular approaches for topic modeling is Latent Dirichlet allocation (LDA) (Blei et al., 2003). LDA provides insights about the latent topical structure of a corpus of documents; consequently, it has been successfully used for clustering documents based on their topics. LDA is a generative model that represent a document in the space of its hidden topics. In other words, in this model lexical variability in documents is explained by their distribution over semantic entities (topics). The name Dirichlet came from generating the topic distribution for a document from a Dirichlet prior. In this model, each word in a document is sampled from the multinomial distribution of a topic where this topic is also generated from the multinomial topic distribution for the document. The main difference between LDA and PLSI is the Dirichlet prior for topic distribution.

Topic modeling has been successfully used in several text mining tasks such as biased language detection in news and blogs (Ahmed and Xing, 2010; Nguyen et al., 2013), depression linguistic signal identification in tweets (Resnik et al., 2015), and contentious expression and interaction mining (Mukherjee and Liu, 2012). Furthermore, it was employed for automatic identification of argument structure in formal documents of 19th-century philosophical texts (Lawrence et al., 2014). In their approach, LDA was applied on the target corpus, and the resulting topics were elaborated to identify similarities between different propositions.

Topic modeling has been also used in sentiment analysis and opinion mining to simultaneously model the topics/aspects and the sentiments in a text (Titov and McDonald, 2008b; Mei et al., 2007). In Titov and McDonald (2008a), the LDA algorithm is used to jointly model sentiments and topics in an unsupervised approach. Gottipati et al. (2013) proposed a generative topic model to extract low-dimensional representations for debate texts. In their approach, texts are preprocessed by extracting named entities. Later, they used the MPQA sentiment lexicon (Wilson et al., 2005a) to estimate the prior over its term distributions. Similarly, Lin and He (2009) modeled each token as sentiment or topic, using LDA topic modeling and sentiment lexicons.

2.5 Deep Neural Networks for Text Classification

In recent years, deep neural networks have drawn substantial attention in the area of supervised learning. There are two main reasons behind the popularity and the high impact of deep neural networks in various fields such as computer vision and speech recognition:

Deep Learning Concept	**Reference**
Convolutional Networks	Goodfellow et al. (2016), Chapter 9
Autoencoders	Goodfellow et al. (2016), Chapter 14
Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) and other Gated RNNs	Goodfellow et al. (2016), Chapter 10
Sequence-to-sequence attention-based models	Bahdanau et al. (2014)

Table 2.2: The list of references for comprehensive descriptions of deep learning concepts used in this thesis

- The emergence of modern parallel computing architectures providing low-cost and fast computation for a large number of parameters of the deep networks.
- The availability of vast amounts of images, video, speech and text on the Internet providing sufficient data for training these networks.

Deep networks are multilayer networks on top of each other where each layer corresponds to a different level of abstraction. Each layer in the deep architecture provides a nonlinear information processing. In these deep networks, the output is a parameterized function of the inputs and the output of each layer is the input for the higher layer. In the learning process, these parameters are optimized. To optimize parameters, gradient-based approaches such as Stochastic Gradient Descent (SGD) are employed, as the error back propagates from final network output down to the input representation. We refer the reader to comprehensive description of deep neural networks in Goodfellow et al. (2016), part II: Modern Practical Deep Networks. Table 2.2 shows the list of references for comprehensive description of deep learning concepts used in the this thesis.

Deep neural networks have been successfully applied for different NLP task including: sentiment analysis (Socher et al., 2011b), machine translation (Sutskever et al., 2014), text generation (Sutskever et al., 2011), part-of-speech tagging (dos Santos and Zadrozny, 2014), named entity recognition (Collobert and Weston, 2008), question answering (Hermann et al., 2015), semantic role labeling (Zhou and Xu, 2015) and automatic parsing (Legrand and Collobert, 2015).

The most common deep learning approaches for text classification are Convolutional and Recurrent Neural Networks (NN). A Convolutional Neural Network (CNN) is a kind of feed-

forward NN that has shown remarkable performance for computer vision. The key feature of CNN is its pooling mechanism which makes it invariant to location or time. However, in natural languages, the location of the word in the sentence plays a significant role in the overall meaning. If we consider a text as a sequence of words or characters, Recurrent Neural Networks (RNN) are more powerful in processing variable-length sequences and performing the same task on every element of the sequence and keep the memory of the information processed before. In section 2.5.1, we review RNN models in more depth.

One of the key advantages of deep neural networks for different NLP applications is that these models can be trained through an end-to-end process to learn both input representation and parameters of the neural network. Thus, no more task and domain-specific feature engineering are required. In section 2.5.3, we further review automatic representations learning for different NLP tasks.

2.5.1 Recurrent neural networks

Recurrent Neural Networks (RNN) Elman (1990) are the most popular models for sequential data and have shown promising results in various NLP task. Basically, they are extensions of feed-forward neural models that can handle variable-length inputs. They are particularly beneficial for sequential data, because, unlike a feed-forward network, they share their hidden states across time and keep track of the information processed, as a memory. The sequential history is summarized in a hidden vector. A standard RNN iterates over the input sequence and computes the hidden state s_t at each step as follows:

$$s_t = f(Ux_t, Ws_{t-1}) \tag{2.3}$$

where f is a non-linear function such as sigmoid, x_t is the current input, and s_{t-1} is the previous hidden state. The first hidden state is normally initialized to all zeros.

While Vanilla RNN might suffer from the decaying of the gradient, or less frequently, blowing-up of gradient problems, different architectures of RNN such as Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Unit (GRU) (Cho et al., 2014a) can capture long-term dependencies by having different gating mechanisms. Even though there are differences between these two types of gated recurrent neural networks, various works (Bahdanau et al., 2014; Chung et al., 2014) reported comparable performance for them.

LSTM replaces the hidden vector of a recurrent neural network with *memory blocks* which are equipped with gates; it can, in principle, keep a long-term memory by training proper gating

weights (refer to Graves (2008) for intuitive illustrations and good discussions). The specific memory copying and gating configurations in LSTM's memory blocks render an effective mechanism in capturing both short and distant interplays in an input sequence. Similarly, GRU has gating units that control the flow of information into the unit but without having separate memory cells Chung et al. (2014).

LSTM has practically shown to be very useful, achieving the state of the art on a range of problems. In Graves et al. (2013), a deep LSTM network achieved the state-of-the-art results on the TIMIT phoneme recognition benchmark. In (Liwicki et al., 2007) and (Graves, 2012), LSTM networks were found to be very useful for digit writing recognition, because of the network's capability of memorizing context information in a long sequence. In Eck and Schmidhuber (2002), LSTM networks were trained to effectively capture the global structures of the temporal data. With the memory cells, LSTM is able to keep track of temporally distant events that indicate global music structures. As a result, LSTM can be successfully trained to compose music, where other RNNs have failed to do so.

RNN was later extended as bi-directional RNN (Irsoy and Cardie, 2014b; Schuster and Paliwal, 1997) as the output at each time step not only depends on previous elements in the sequence but also might depend on the next elements in the sequence. A bi-directional RNN consists of two RNNs, one that processes the input in its original order and one that processes the reversed input sequence. Finally, the output is the function of hidden states of both RNNs. For more details about sequence modeling and in particular RNNs, we refer the reader to Goodfellow et al. (2016), Chapter 10.

2.5.2 Sequence to Sequence Learning with Deep Neural Models

Encoder-decoder sequence-to-sequence models (Sutskever et al., 2014; Cho et al., 2014b) were originally used for machine translation, where a bi-directional RNN is trained to learn the representation for the source language and the other RNN generates the translation in the target language. In these works, the source-language sentences are encoded with LSTM as a real-valued vector, which is, in turn, used to initialized the LSTM used to decode the target sentences, in which the *meaning* of the source side sentences is translated.

These neural models were later extended for tasks with variable input and output sequence length including: end-to-end neural machine translation (Sutskever et al., 2014; Cho et al., 2014b), image-to-text conversion (Vinyals et al., 2015b), syntactic constituency parsing (Vinyals et al., 2015a) and question answering (Hermann et al., 2015). Subsequently, the attention mechanism allowed the models to learn alignments between different parts of the source

and the target such as between speech frames and its text in speech recognition (Chorowski et al., 2014) or between image frames and agent actions in the dynamic control problem (Mnih et al., 2014).

2.5.3 Word Vector Representations

The main advantage of deep networks for NLP applications emerges from the advent of word embeddings (Irsoy and Cardie, 2014a). Word embeddings are low-dimensional real-valued vectors used to represent words in the vocabulary in the meaning space Bengio et al. (2001). The 'low' dimensionality is relative to the vocabulary size, and using a few hundred dimensions is common. A number of different language modeling techniques have been proposed to generate word embeddings, all of which require only a large corpus of text (e.g., Collobert and Weston (2008); Mnih and Hinton (2009)).

These word vectors have been demonstrated to be beneficial for different NLP problems (Soricut and Och, 2015; Mnih and Hinton, 2007; Socher et al., 2011a; Faruqui et al., 2014; Soricut and Och, 2015), as they can capture different syntactic and semantic properties of natural language (Mikolov et al., 2013c). They have been particularly used as extra features in a statistical learner for a number of tasks including sentiment analysis Tang et al. (2014) and named entity recognition Turian et al. (2010).

The key advantage of word embeddings is that they can be learned in an unsupervised manner from the distributional information of words in a large corpus (Mikolov et al., 2013a; Pennington et al., 2014), as well as by supervised approaches for particular tasks (Socher et al., 2013; Tang et al., 2014). For instance, in sentiment analysis, the embeddings that are trained solely based on distributional properties of words are problematic, as they may map opposite words like "good" and "bad" to similar vectors because they appear in the similar syntactic contexts. Therefore, some mechanisms are required to incorporate supervision for determining the sentiment of a text (Tang et al., 2014). Another benefit of using word embedding for NLP tasks is that they diminish the need for feature engineering and hand-crafted features.

2.6 Summary

In this chapter, several related works to our research were reviewed. We started by sentiment analysis and opinion mining approaches, as stance classification is similar and arguably complementary to these tasks. We continued by exploring previous works for stance classification and argument mining, as they are the main focus of this thesis. As we employ topic modeling

and deep learning approaches to detect stance and the reasons behind it in user-generated texts, we studied these methods in more details and investigated their efficiency in various NLP tasks.

In the next chapters, we describe our methods for detecting stance in Twitter data and news comments. Additionally, new datasets that created and manually annotated for stance detection are introduced.

Chapter 3

Detecting Stance in Tweets and Analyzing its Interaction with Sentiment

3.1 Introduction

Stance detection is the task of automatically determining from text whether the author of the text is in favor of, against, or neutral towards a proposition or a target. The target may be a person, an organization, a government policy, a movement, a product, etc. For example, one can infer from Barack Obama's speeches that he is in favor of stricter gun laws in the US. Similarly, people often express stance towards various target entities through posts on online forums, blogs, Twitter, Youtube, Instagram, etc.

Automatically detecting stance has widespread applications in information retrieval, text summarization, and textual entailment. Over the last decade, there has been active research in modeling stance. However, most work focuses on congressional debates Thomas et al. (2006) or debates in online forums Somasundaran and Wiebe (2010); Anand et al. (2011); Walker et al. (2012b); Hasan and Ng (2013). Here we explore the task of detecting stance in Twitter—a popular microblogging platform where people often express stance implicitly or explicitly.

The task we explore is formulated as follows: given a tweet text and a target entity (person, organization, issue, etc.), automatic natural language systems must determine whether the tweeter is in favor of the given target, against the given target, or whether neither inference is likely. For example, consider the target–tweet pair:

Target: legalization of abortion (1)
Tweet: *The pregnant are more than walking incubators. They have rights too!*

Humans can deduce from the tweet that the tweeter is likely in favor of the target.[1]

Note that lack of evidence for 'favor' or 'against', does not imply that the tweeter is neutral towards the target. It may just mean that we cannot deduce stance from the tweet. In fact, this is a common phenomenon. On the other hand, the number of tweets from which we can infer neutral stance is expected to be small. Example:

Target: Hillary Clinton (2)
Tweet: *Hillary Clinton has some strengths and some weaknesses.*

Stance detection is related to, but different from, sentiment analysis. Sentiment analysis tasks are formulated as determining whether a piece of text is positive, negative, or neutral, or determining from the text the speaker's opinion and the target of the opinion (the entity towards which opinion is expressed). However, in stance detection, systems are to determine favorability towards a given (pre-chosen) target of interest. The target of interest may not be explicitly mentioned in the text or it may not be the target of opinion in the text. For example, consider the target–tweet pair below:

Target: *Donald Trump* (3)
Tweet: *Jeb Bush is the only sane candidate in this republican lineup.*

The target of opinion in the tweet is Jeb Bush, but the given target of interest is Donald Trump. Nonetheless, we can infer that the tweeter is likely to be unfavorable towards Donald Trump. Also note that, in stance detection, the target can be expressed in different ways which impacts whether the instance is labeled 'favor' or 'against'. For example, the target in example 1 could have been phrased as 'pro-life movement', in which case the correct label for that instance is 'against'. Also, the same stance ('favor' or 'against') towards a given target can be deduced from positive tweets and negative tweets. This interaction between sentiment and stance has not been adequately addressed in past work, and an important reason for this is the lack of a dataset annotated for both stance and sentiment.
The contributions of this chapter are as follows, based on the following publications:

1. *Created a new stance dataset*: We created the first dataset of tweets labeled for both stance and sentiment (Section 3.2 and Section 3.3). More than 4,000 tweets are annotated for whether one can deduce favorable or unfavorable stance towards one of five targets 'Atheism', 'Climate Change is a Real Concern', 'Feminist Movement', 'Hillary Clinton', and 'Legalization of Abortion'. Each of these tweets is also annotated for whether the target of opinion expressed in the tweet is the same as the given target of interest. Finally,

[1]Note that we use 'tweet' to refer to the text of the tweet and not to its meta-information.

each tweet is annotated for whether the given text conveys the positive, negative, or neutral sentiment. The construction of the dataset is described in:

- Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. A dataset for detecting stance in tweets. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3945–3952, may 2016b

2. *Organized a shared task competition on stance detection*: Partitions of this stance-annotated data were used as a training and test sets in the SemEval-2016 shared task competition, Task #6: Detecting Stance from Tweets. Participants were provided with 2,914 training instances labeled for stance for the five targets. The test data included 1,249 instances. All of the stance data is made freely available through the shared task website. The task received submissions from 19 teams. The best performing system obtained an overall average F-score of 67.8. Their approach employed two recurrent neural networks (RNN) classifiers: the first was trained to predict task-relevant hashtags on a large unlabeled Twitter corpus. This network was used to initialize a second RNN classifier, which was trained with the provided training data Zarrella and Marsh (2016). The overview of the shared task is described in the following publication:

   - Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. SemEval-2016 Task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, San Diego, California, 2016c

3. *Developed a state-of-the-art stance detection system*: We propose a stance detection system that is much simpler than the shared task winning system (described above), and yet obtains an even better F-score of 70.3 on the shared task's test set (Sections 3.5, 3.6 and 3.7). We use a linear-kernel SVM classifier that relies on features drawn from the training instances—such as word and character n-grams—as well as those obtained using external resources—such as word-embedding features from additional unlabeled data.

   - Parinaz Sobhani, Saif Mohammad, and Svetlana Kiritchenko. Detecting stance in tweets and analyzing its interaction with sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 159–169, Berlin, Germany, August 2016

4. *Explored several research questions*: We conduct several experiments to better understand stance detection and its interaction with sentiment (Section 3.6).

   - Question: What is the extent to which stance can be determined simply by traditional sentiment analysis (identifying positive and negative language)?

     Experiment: We use gold sentiment labels to determine stance, and compare results with several baselines. We show that even though determining sentiment helps determine stance to some extent (leads to results higher than majority baseline), it is not sufficient.

   - Question: How useful are sentiment analysis features for stance detection? How does the usefulness of the same features vary when determining stance vs. when determining sentiment?

     Experiment: We apply the stance detection system (mentioned above in (4)), as a common text classification framework, to determine both stance and sentiment. We show that while sentiment features are markedly useful for sentiment classification, they are not as effective for stance classification. Furthermore, even though both stance and sentiment detection are framed as three-way classification tasks on a common dataset, automatic systems perform markedly better when detecting sentiment than when detecting stance towards a given target.

   - Question: How much does the performance of a stance classification system vary on instances where the target of interest is also the target of opinion vs. instances where the target of interest is different from the target of opinion?

     Experiment: We show that stance detection is particularly challenging when the tweeter expresses an opinion about an entity other than the target of interest. (The text classification system performs close to majority baseline for such instances.)

This chapter is largely based on:

Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. *Special Section of the ACM Transactions on Internet Technology on Argumentation in Social Media*, In Press, 2016d

3.2 A Dataset for Stance from Tweets

We now explain how we compiled a set of tweets and targets for stance annotation (Section 3.2.1), and the questionnaire and crowdsourcing setup used for stance annotation (Section 3.2.2). An analysis of the stance annotations is presented in Section 3.4.

3.2.1 Selecting the Tweet–Target Pairs

Our goal was to create a stance-labeled dataset with the following properties:

1. The tweet and target are commonly understood by a wide number of people in the US. (The data was eventually annotated for stance by respondents living in the US.)

2. There must be a significant amount of data for the three classes: 'favor', 'against', and 'neither'.

3. Apart from tweets that explicitly mention the target, the dataset should include a significant number of tweets that express opinion towards the target without referring to it by name. We wanted to include the relatively harder cases for stance detection where the target is referred to in indirect ways such as through pronouns, epithets, honorifics, and relationships.

4. Apart from tweets that express opinion towards the target, the dataset should include a significant number of tweets in which the target of opinion is different from the given target of interest. Downstream applications often require stance towards particular pre-chosen targets of interest (for example, a company might be interested in stance towards its product). Having data where the target of opinion is some other entity (for example, a competitor's product), helps test how well stance detection systems can cope with such instances.

These properties influenced various choices in how our dataset was created. To help with Property 1, we compiled a list of target entities commonly known in the United States: 'Atheism', 'Climate Change is a Real Concern", 'Feminist Movement', 'Hillary Clinton', and 'Legalization of Abortion'.

We created a small list of hashtags, which we will call *query hashtags*, that people use when tweeting about the targets. We split these hashtags into three categories: (1) *favor hashtags*: expected to occur in tweets expressing favorable stance towards the target (for example, *#Hillary4President*), (2) *against hashtags:* expected to occur in tweets expressing opposition to the target (for example, *#HillNo*), and (3) *stance-ambiguous hashtags:* expected to occur in tweets about the target, but are not explicitly indicative of stance (for example, *#Hillary2016*).[2]

[2]A tweet that has a seemingly favorable hashtag towards a target may in fact oppose the target; and this is not uncommon. Similarly, unfavorable (or against) hashtags may occur in tweets that favor the target.

Target	Example Favor Hashtag	Example Against Hashtag	Example Ambiguous Hashtag
Atheism	*#NoMoreReligions*	*#Godswill*	*#atheism*
Climate Change is Concern	-	*#globalwarminghoax*	*#climatechange*
Feminist Movement	*#INeedFeminismBecaus*	*#FeminismIsAwful*	*#Feminism*
Hillary Clinton	*#GOHILLARY*	*#WhyIAmNotVotingForHillary*	*#hillary2016*
Legalization of Abortion	*#proChoice*	*#prayToEndAbortion*	*#PlannedParenthood*

Table 3.1: Examples of stance-indicative and stance-ambiguous hashtags that were manually identified.

We will refer to favor and against hashtags jointly as *stance-indicative (SI) hashtags*. Table 3.1 lists some of the hashtags used for each of the targets. (We were not able to find a hashtag that is predominantly used to show favor towards 'Climate change is a real concern', however, the stance-ambiguous hashtags were the source of a large number of tweets eventually labeled 'favor' through human annotation.) Next, we polled the Twitter API to collect over two million tweets containing these query hashtags. We kept only those tweets where the query hashtags appeared at the end. We removed the query hashtags from the tweets to exclude obvious cues for the classification task. Since we only select tweets that have the query hashtag at the end, removing them from the tweet often does not impact the grammaticality of the original tweet.

Note that the presence of a stance-indicative hashtag is not a guarantee that the tweet will have the same stance. Further, removal of query hashtags may result in a tweet that no longer expresses the same stance as with the query hashtag. Thus we manually annotate the tweet–target pairs after the pre-processing described above. For each target, we sampled an equal number of tweets pertaining to the favor hashtags, the against hashtags, and the stance-ambiguous hashtags—up to 1000 tweets at most per target. This helps in obtaining a sufficient number of tweets pertaining to each of the stance categories (Property 2). Properties 3 and 4 are addressed to some extent by the fact that removing the query hashtag can sometimes result in tweets that do not explicitly mention the target. Consider:

Target: Hillary Clinton (4)
Tweet: *Benghazi must be answered for #Jeb16*

The query hashtags '#HillNo' was removed from the original tweet, leaving no mention of Hillary Clinton. Yet there is sufficient evidence (through references to Benghazi and #Jeb16) that the tweeter is likely against Hillary Clinton. Further, conceptual targets such as 'Legalization of Abortion' (much more so than person-name targets) have many instances where the target is not explicitly mentioned. For example, tweeters can express stance by referring to *foetuses, women's rights, freedoms*, etc., without having to mention *legalization* or *abortion*.

3.2.2 Stance Annotation

The instructions given to annotators for determining stance are shown below. In our annotation task, we asked respondents to label for stance towards a given target based on the tweet text alone. However, automatic systems may benefit from exploiting tweet meta-information. Descriptions within each option make clear that stance can be expressed in many different ways, for example by explicitly supporting or opposing the target, by supporting an entity aligned with or opposed to the target, etc. The second question asks whether the target of opinion in the tweet is the same as the given target of interest.

Target of Interest: [target entity]
Tweet: [tweet with query hashtag removed]

Q1: From reading the tweet, which of the options below is most likely to be true about the tweeter's stance or outlook towards the target:

- We can infer from the tweet that the tweeter supports the target

  This could be because of any of reasons shown below:

  - *the tweet is explicitly in support for the target*
  - *the tweet is in support of something/someone aligned with the target, from which we can infer that the tweeter supports the target*
  - *the tweet is against something/someone other than the target, from which we can infer that the tweeter supports the target*
  - *the tweet is NOT in support of or against anything, but it has some information, from which we can infer that the tweeter supports the target*
  - *we cannot infer the tweeters stance towards the target, but the tweet is echoing somebody else's favorable stance towards the target (in a news story, quote, retweet, etc.)*

- We can infer from the tweet that the tweeter is against the target

  This could be because of any of the following:

  - *the tweet is explicitly against the target*
  - *the tweet is against someone/something aligned with the target entity, from which we can infer that the tweeter is against the target*
  - *the tweet is in support of someone/something other than the target, from which we can infer that the tweeter is against the target*
  - *the tweet is NOT in support of or against anything, but it has some information, from which we can infer that the tweeter is against the target*
  - *we cannot infer the tweeters stance towards the target, but the tweet is echoing somebody else's negative stance towards the target entity (in a news story, quote, retweet, etc.)*

- We can infer from the tweet that the tweeter is neutral towards the target
  The tweet must provide some information that suggests that the tweeter is neutral towards the target – the tweet being neither favorable nor against the target is not sufficient reason for choosing this
- There is no clue in the tweet to reveal the stance of the tweeter towards the target (support/against/neutral)

Q2: From reading the tweet, which of the options below is most likely to be true about the focus of opinion/sentiment in the tweet:

- The tweet explicitly expresses opinion about the target
- The tweet expresses opinion about something/someone other than the target
- The tweet is not expressing opinion about anything

Each of the tweet–target pairs selected for annotation was uploaded on CrowdFlower for annotation with the questionnaire shown above.[3] We used CrowdFlower's gold annotations scheme for quality control, wherein about 5% of the data was annotated internally (by the authors). These questions are referred to as gold questions. During crowd annotation, the gold questions are interspersed with other questions, and the annotator is not aware which is which. However, if she gets a gold question wrong, she is immediately notified of it. If the accuracy of the annotations on the gold questions falls below 70%, the annotator is refused further annotation. This serves as a mechanism to avoid malicious annotations. In addition, the gold questions serve as examples to guide the annotators.

Each question was answered by at least eight respondents. The respondents gave the task high scores in a post-annotation survey despite noting that the task itself requires some non-trivial amount of thought: 3.9 out of 5 on ease of task, 4.4 out of 5 on the clarity of instructions, and 4.2 out of 5 overall.

For each target, the data not annotated for stance is used as the *domain corpus*—a set of unlabeled tweets that can be used to obtain information helpful to determine stance, such as relationships between relevant entities (we explore the use of the domain corpus in Section 3.7).

The number of instances that were marked as neutral stance (option 3 in question 1) was less than 1%. Thus, we merged options 3 and 4 into one 'neither in favor nor against' option ('neither' for short). The inter-annotator agreement was 73.1% for question 1 (stance)

[3] http://www.crowdflower.com

and 66.2% for Question 2 (target of opinion).[4] These statistics are for the complete annotated dataset, which includes instances that were genuinely difficult to annotate for stance (possibly because the tweets were too ungrammatical or vague) and/or instances that received poor annotations from the crowd workers (possibly because the particular annotator did not understand the tweet or its context). In order to aggregate stance annotation information from multiple annotators for an instance, rather than opting for simple majority, we marked an instance with a stance only if at least 60% of the annotators agreed with each other; the instances with less than 60% agreement were set aside.[5] We will refer to this dataset of 4,163 instances as the *Stance Dataset*. The inter-annotator agreement on this Stance Dataset is 81.85% for question 1 (stance) and 68.9% for Question 2 (target of opinion).

3.3 Labeling the Stance Set for Sentiment

A key research question this work aims at addressing is the extent to which sentiment is correlated with the stance. To that end, we annotated the same Stance Dataset described above for sentiment in a separate annotation effort a few months later. We followed a procedure for annotation on CrowdFlower similar to that described above for stance, but now provided only the tweet (no target).

Prior work in sentiment annotation has often simply asked the annotator to label a sentence as positive, negative, or neutral, largely leaving the notion when pieces of text should be marked as positive, negative, or neutral up to the individual annotators. This is problematic because it can lead to differing annotations from annotators for the same text. Further, in several scenarios the annotators may be unsure about how best to label the text. Some of these scenarios are listed below. (See Mohammad (2016) for further discussion on the challenges of sentiment annotation.)

- *Sarcasm and ridicule*: Sarcasm and ridicule are tricky from the perspective of assigning a single label of sentiment because they can often indicate the positive emotional state of the speaker (pleasure from mocking someone or something) even though they have

[4]The overall inter-annotator agreement was calculated by averaging the agreements on all tweets in the dataset. For each tweet, the inter-annotator agreement was calculated as the number of annotators who agree over the majority label divided by the total number of annotators for that tweet.

[5]The 60% threshold is somewhat arbitrary, but it seemed appropriate in terms of balancing confidence in the majority annotation and having to discard too many instances. Annotations for 28% of the instances did not satisfy this criterion. Note that even though we request 8 annotations per questions, some questions may be annotated more than 8 times. Also, a small number of instances received less than 8 annotations.

a negative attitude towards someone or something. An example of ridicule from our dataset:

> *DEAR PROABORTS: Using BAD grammar and FILTHY language and INTIMIDATION makes you look ignorant, inept and desperate. #GodWins*

- *Supplications and requests*: Many tweets convey positive supplications to God or positive requests to people in the context of a (usually) negative situation. Example from our dataset:

  > *Pray for the Navy yard. God please keep the casualties minimal. #1A #2A #NRA #COS #CCOT #TGDN #PJNET #WAKEUPAMERICA*

- *Rhetorical questions*: Rhetorical questions can be treated simply as queries (and thus neutral) or as utterances that give away the emotional state of the speaker. For example, consider this example from our dataset of tweets:

  > *How soon do you think WWIII &WWWIV will begin? #EndRacism*

  On the one hand, this tweet can be treated as a neutral question, but on the other hand, it can be seen as negative because the utterance betrays a sense of frustration on the part of the speaker.

After a few rounds of internal development, we used the questionnaire below to annotate for sentiment:

What kind of language is the speaker using?

1. the speaker is using positive language, for example, expressions of support, admiration, positive attitude, forgiveness, fostering, success, positive emotional state (happiness, optimism, pride, etc.)
2. the speaker is using negative language, for example, expressions of criticism, judgment, negative attitude, questioning validity/competence, failure, negative emotional state (anger, frustration, sadness, anxiety, etc.)
3. the speaker is using expressions of sarcasm, ridicule, or mockery
4. the speaker is using positive language in part and negative language in part
5. the speaker is neither using positive language nor using negative language

---

The use of the phrases ‘positive language’ and ‘negative language’ encourages respondents to focus on the language itself as opposed to assigning sentiment based on event outcomes

			% of instances in Train				% of instances in Test		
Target	**# total**	**# train**	favor	against	neither	**# test**	favor	against	neither
Atheism	733	513	17.9	59.3	22.8	220	14.5	72.7	12.7
Climate Change	564	395	53.7	3.8	42.5	169	72.8	6.5	20.7
Feminist Movement	949	664	31.6	49.4	19.0	285	20.4	64.2	15.4
Hillary Clinton	983	689	17.1	57.0	25.8	295	15.3	58.3	26.4
Legal. Abortion	933	653	18.5	54.4	27.1	280	16.4	67.5	16.1
Total	4163	2914	25.8	47.9	26.3	1249	23.1	51.8	25.1

Table 3.2: Distribution of instances in the Stance Train and Test sets for Question 1 (Stance).

that are beneficial or harmful to the annotator. Sarcasm, ridicule, and mockery are included as a separate option (in addition to option 2) so that respondents do not have to wonder if they should mark such instances as positive or negative. Instances with different sentiment towards different targets of opinion can be marked with option 4. Supplications and requests that convey a sense of fostering and support can be marked as positive. On the other hand, rhetorical questions that betray a sense of frustration and disappointment can be marked as negative.

Each instance was annotated by at least five annotators on CrowdFlower. The respondents gave the task high scores in a post-annotation survey: 4.2 out of 5 on ease of task, 4.4 out of 5 on the clarity of instructions, and 4.2 out of 5 overall.

For our current work, we chose to combine options 2 and 3 into one 'negative tweets' class but they can be kept separate in future work if so desired. We also chose to combine options 4 and 5 into one 'neither clearly positive nor clearly negative category' ('neither' for short). This frames the automatic sentiment prediction task as a three-way classification task, similar to the stance prediction task. The inter-annotator agreement on the sentiment responses across these three classes was 85.6%.

3.4 Properties of the Stance Dataset

We partitioned the Stance Dataset into training and test sets based on the timestamps of the tweets. For each target, the annotated tweets were ordered by their timestamps, and the first 70% of the tweets formed the training set and the last 30% formed the test set. Table 5.1 shows the number and distribution of instances in the Stance Dataset.

Table 3.3 shows the distribution of responses to Question 2 (whether the opinion is ex-

	Opinion towards		
Target	Target	Other	No one
Atheism	49.25	46.38	4.37
Climate Change is Concern	60.81	30.50	8.69
Feminist Movement	68.28	27.40	4.32
Hillary Clinton	60.32	35.10	4.58
Legalization of Abortion	63.67	30.97	5.36
Total	61.02	33.77	5.21

Table 3.3: Percentage distribution of instances in the Stance Dataset (4163 training and test instances) for Question 2.

	Opinion towards		
Stance	Target	Other	No one
favor	94.23	5.11	0.66
against	72.75	26.54	0.71

Table 3.4: Percentage distribution of instances by target of opinion across stance labels in the Stance Dataset (4163 training and test instances).

pressed directly about the given target). Observe that the percentage of 'opinion towards other' varies across different targets from 27% to 46%. Table 3.4 shows the distribution of instances by the target of opinion for the 'favor' and 'against' stance labels. Observe that, as in Example 3, in a number of tweets from which we can infer unfavorable stance towards a target, the target of opinion is someone/something other than the target (about 26.5%). Manual inspection of the data also revealed that in a number of instances, the target is not directly mentioned, and yet stance towards the target was determined by the annotators. About 28% of the 'Hillary Clinton' instances and 67% of the 'Legalization of Abortion' instances were found to be of this kind—they did not mention 'Hillary' or 'Clinton' and did not mention 'abortion', 'pro-life', and 'pro-choice', respectively (case insensitive; with or without hashtag; with or without a hyphen). Examples (1) and (4) shown earlier are instances of this and are taken from our dataset. Some other examples are shown below:

Target: Hillary Clinton (5)
Tweet: *I think I am going to vote for Monica Lewinsky's Ex-boyfriends Wife*

Target: Legalization of Abortion (6)
Tweet: *The woman has a voice. Who speaks for the baby? I'm just askin.*

	% of instances in Train			% of instances in Test		
Target	positive	negative	neither	positive	negative	neither
Atheism	60.43	35.09	4.48	59.09	35.45	5.45
Climate Change is Concern	31.65	49.62	18.73	29.59	51.48	18.93
Feminist Movement	17.92	77.26	4.82	19.30	76.14	4.56
Hillary Clinton	32.08	64.01	3.92	25.76	70.17	4.07
Legalization of Abortion	28.79	66.16	5.05	20.36	72.14	7.5
Total	33.05	60.47	6.49	29.46	63.33	7.20

Table 3.5: Distribution of instances by sentiment in the Stance Train set (total 2914 instances) and Test set (total 1249 instances).

Table 3.5 shows the distribution of sentiment labels in the training and test sets. Observe that tweets corresponding to all targets, except for 'Atheism', are predominantly negative.

3.5 A Common Text Classification Framework for Stance and Sentiment

Past works have shown that the most useful features for sentiment analysis are word and character n-grams and sentiment lexicons, whereas others such as negation features, part-of-speech features, and punctuation have a smaller impact (Wilson et al., 2013a; Mohammad et al., 2013b; Kiritchenko et al., 2014b; Rosenthal et al., 2015). These features may be useful in stance classification as well; however, it is unclear which features will be more useful (and to what extent). Since we now have a dataset annotated for both stance and sentiment, we create a common text classification system (common machine learning framework and common features) and apply it to the Stance Dataset for both stance and sentiment classification.

There is one exception to the common machine learning framework. The words and concepts used in tweets corresponding to the three stance categories are not expected to generalize across the targets. Thus, the stance system learns a separate model from training data pertaining to each of the targets.[6] Positive and negative language tend to have sufficient amount of commonality regardless of the topic of discussion, and hence sentiment analysis systems traditionally learn a single model from all of the training data (Liu, 2015; Kiritchenko et al., 2014b; Rosenthal et al., 2015). Thus, our sentiment experiments are also based on a single

[6]Experiments with a stance system that learns a single model from all training tweets showed lower results.

model trained on all of the Stance Training set.

Tweets are tokenized and part-of-speech tagged with the CMU Twitter NLP tool (Gimpel et al., 2011). We train a linear-kernel Support Vector Machine (SVM) classifier on the Stance Training set. SVMs have proven to be effective on text categorization tasks and robust on large feature spaces. We use the SVM implementation provided by the scikit-learn Machine Learning library (Pedregosa et al., 2011).
The features used in our text classification system are shown below:

- n-*grams*: presence or absence of contiguous sequences of 1, 2 and 3 tokens (word n-grams); presence or absence of contiguous sequences of 2, 3, 4, and 5 characters (character n-grams);

- *sentiment (sent.)*: The sentiment lexicon features are derived from three manually created lexicons: NRC Emotion Lexicon Mohammad and Turney (2010), Hu and Liu Lexicon Hu and Liu (2004), and MPQA Subjectivity Lexicon Wilson et al. (2005b), and two automatically created, tweet-specific, lexicons: NRC Hashtag Sentiment and NRC Emoticon (a.k.a. Sentiment140) lexicons Kiritchenko et al. (2014b);

- *target*: presence/absence of the target of interest in the tweet;[7]

- *POS*: the number of occurrences of each part-of-speech tag (POS);

- *encodings (enc.)*: presence/absence of positive and negative emoticons, hashtags, characters in upper case, elongated words (e.g., *sweeettt*), and punctuations such as exclamation and question marks.

The SVM parameters are tuned using 5-fold cross-validation on Stance Training set. We evaluate the learned models on the Stance Test set. As the evaluation measure, we use the average of the F1-scores (the harmonic mean of precision and recall) for the two main classes:[8]

For stance classification:

$$F_{average} = \frac{F_{favor} + F_{against}}{2} \tag{3.1}$$

For sentiment classification:

$$F_{average} = \frac{F_{positive} + F_{negative}}{2} \tag{3.2}$$

[7]For instance, for 'Hillary Clinton' the mention of either 'Hillary' or 'Clinton' (case insensitive; with or without hashtag) in the tweet shows the presence of target.

[8]A similar metric was used in the past for sentiment analysis—SemEval 2013 Task 2 Wilson et al. (2013a).

Classifier	Atheism	Climate Change	Feminist Movemt.	Hillary Clinton	Legal. of Abortion	F-macroT	F-microT
I. Benchmarks							
a. Random	31.1	27.8	29.1	33.5	31.1	30.5	33.3
b. Majority	42.1	42.1	39.1	36.8	40.3	40.1	65.2
c. First in shared task	61.4	41.6	62.1	57.7	57.3	56.0	67.8
d. Oracle Sentiment	65.8	34.3	61.7	62.2	41.3	53.1	57.2
e. Oracle Sentiment and Target	66.2	36.2	63.7	72.5	41.8	56.1	59.6
II. Our SVM classifier							
a. n-grams	65.2	**42.4**	57.5	58.6	**66.4**	58.0	69.0
b. a. + POS	**65.8**	41.8	**58.7**	57.6	62.6	57.3	68.3
c. a. + encodings	65.7	42.1	57.6	58.4	64.5	57.6	68.6
d. a. + target	65.2	42.2	57.7	60.2	66.1	**58.3**	**69.1**
e. a. + sentiment	65.2	40.1	54.5	**60.6**	61.7	56.4	66.8

Table 3.6: Stance Classification: F-scores obtained for each of the targets (the columns) by the benchmark systems and our classifier. Macro- and micro-averages across targets are also shown. The highest scores are shown in bold.

Note that $F_{average}$ can be determined for all of the test instances or for each target data separately. We will refer to the $F_{average}$ obtained through the former method as *F-micro-across-targets* or *F-microT* as a shorter notation. On the other hand, the $F_{average}$ obtained through the latter method, that is, by averaging the $F_{average}$ calculated for each target separately, will be called *F-macro-across-targets* or *F-macroT* as a shorter notation. Systems that perform relatively better on the more frequent target classes will obtain higher F-microT scores. On the other hand, to obtain a high F-macroT score a system has to perform well on all target classes.

Note that this measure does not give any credit for correctly classifying 'neither' instances. Nevertheless, the system has to correctly predict all three classes to avoid being penalized for misclassifying 'neither' instances as 'favor' or 'against'.

3.6 Results Obtained by Automatic Systems

We now present results obtained by the classifiers described above on detecting stance and sentiment on the Stance Test set. In this section, we focus on systems that use only the provided training data and existing resources such as sentiment lexicons. In Section 3.7, we conduct experiments with systems that use additional unlabeled (or pseudo-labeled) tweets as well.

3.6.1 Results for Stance Classification

We conducted 5-fold cross-validation on the Stance Training set to determine the usefulness of each of the features discussed above. Experiments on the test set showed the same patterns. Due to space constraints, we show results only on the test set — Table 3.6. Rows I.a. to I.e. present benchmarks. Row I.a. shows results obtained by a random classifier (a classifier that randomly assigns a stance class to each instance), and Row I.b. shows results obtained by the majority classifier (a classifier that simply labels every instance with the majority class).[9] Observe that the F-microT for the majority classifier is rather high. This is mostly due to the differences in the class distributions for the five targets: for most of the targets, the majority of the instances are labeled as 'against' whereas for target 'Climate Change is a Real Concern' most of the data are labeled as 'favor'. Therefore, the F-scores for the classes 'favor' and 'against' are more balanced over all targets than for just one target. Row I.c. shows results obtained by MITRE, the winning system (among nineteen participating teams) in the 2016 SemEval shared task on this data (Task #6) (Zarrella and Marsh, 2016).

Results of Oracle Sentiment Benchmarks:
The Stance Dataset with labels for both stance and sentiment allows us, for the first time, to conduct an experiment to determine the extent to which stance detection can be solved with sentiment analysis alone. Specifically, we determine the performance of an oracle a system that assigns stance as follows: For each target, select a sentiment-to-stance assignment (mapping all positive instances to 'favor' and all negative instances to 'against' OR mapping all positive instances to 'against' and all negative instances to 'favor') that maximizes the F-macroT score.[10] We call this benchmark the Oracle Sentiment Benchmark. This benchmark is informative because it gives an upper bound of the F-score one can expect when using a traditional sentiment analysis system for stance detection by simply mapping sentiment labels to stance labels.[11]

In our second sentiment benchmark, Oracle Sentiment and Target, we include the information on the target of opinion. Recall that the Stance Dataset is also annotated for whether the target of opinion is the same as the target of interest. We use these annotations as follows: If the target of opinion is the same as the target of interest, the stance label is assigned in the same

[9] Since our evaluation measure is the average of the F1-scores for the 'favor' and 'against' classes, the random benchmark depends on the distribution of these classes and is different for different targets. The majority class is determined separately for each target.

[10] Tweets with sentiment label 'neither' are always mapped to the stance label 'neither'.

[11] This is an upper bound because gold sentiment labels are used and because the sentiment-to-stance assignment is made in a way that is not usually available in real-world scenarios.

way as in the Oracle Sentiment Benchmark; if the target of opinion is some other entity (whose relation to the target of interest we do not know), we select the sentiment-to-stance assignment from the three options: mapping all positive instances to 'favor' and all negative instances to 'against' OR mapping all positive instances to 'against' and all negative instances to 'favor' OR mapping all instances to 'neither'. Tweets with no opinion are assigned the 'neither' class. Again, the selection is done as to optimize the F-macroT score. This benchmark indicates the level of performance one can expect when a sentiment analysis system is supplemented with the information on the target of opinion.

Rows I.d. and I.e. in Table 3.6 show the F-scores obtained by the Oracle Sentiment Benchmarks on the Stance Test set. Observe that the scores are higher than the majority baseline for most of the targets, but yet much lower than 100%. This shows that even though sentiment can play a key role in detecting stance, sentiment alone is not sufficient.

Results Obtained by Our Classifier:

Row II.a. shows results obtained by our classifier with n-gram features alone. Note that not only are these results markedly higher than the majority baseline, most of these results are also higher than the best results obtained in SemEval-2016 Task 6 (I.c.) and the Oracle benchmarks (I.d. and I.e.). However, character n-grams showed to be considerably more important compared to word n-grams (using only character n-grams we can achieve 56.23 F-macroT). It is mainly because we have relatively small datasets for each target which makes word bigrams and trigrams features very sparse.

Surpassing the best SemEval-2016 results with a simple SVM-ngrams implementation is a little surprising, but it is possible that the SemEval teams did not implement a strong n-gram baseline such as that presented here, or obtained better results using additional features in cross-validation that did not translate to better results when applied to the test set. (The best systems in SemEval-2016 Task 6 used recurrent neural networks and word embeddings.)

Rows II.b. through II.e. show results obtained when using other features (one at a time) over and above the n-gram features. Observe that adding the target features leads to small improvements, but adding all other features (including those drawn from sentiment lexicons) does not improve results. Additional combinations of features such as 'n-grams + target + sentiment' also did not improve the performance (the results are not shown here).

Table 3.7 shows stance detection F-scores obtained by our classifier (SVM with n-gram and target features) over the subset of tweets that express opinion towards the given target and the subset of tweets that express opinion towards another entity.[12] Observe that the performance

[12]The results for the Oracle Sentiment and Target benchmark are low on the subset of tweets that express

	F-macroT		F-microT	
Classifier	To Target	To Other	To Target	To Other
Benchmarks				
a. Random	34.3	20.0	37.4	21.6
b. Majority	44.1	28.6	71.2	41.3
c. First in shared task	59.7	35.4	72.5	44.5
d. Oracle Sentiment	61.0	30.0	65.3	33.3
e. Oracle Sentiment and Target	61.0	15.7	65.3	28.8
Our SVM classifier				
a. n-grams + target	62.5	37.9	75.0	43.0

Table 3.7: Stance Classification: F-scores obtained on tweets with opinion towards the target and on tweets with opinion towards another entity.

of the classifier is considerably better for tweets where opinion is expressed towards the target, than otherwise. Detecting stance towards a given target from tweets that express opinion about some other entity has not been addressed sufficiently in our research community, and we hope our dataset will encourage more work to address this challenging task.

3.6.2 Results for Sentiment Classification

Table 3.8 shows F-scores obtained by various automatic systems on the sentiment labels of the Stance Test set. Observe that the text classification system obtains markedly higher scores on sentiment prediction than on predicting stance.

Once again a classifier trained with n-gram features alone obtains results markedly higher than the baselines (II.a.). However, here (unlike as in the stance task) sentiment lexicon features provide marked further improvements (II.d). Adding POS and encoding features over and above n-grams results in modest gains (II.b. and II.c.) Yet, a classifier trained with all features (II.e.) does not outperform the classifier trained with only n-gram and sentiment features (II.d.).

Table 3.9 shows the performance of the sentiment classifier (SVM with n-grams and sentiment features) on tweets that express opinion towards the given target and those that express an opinion about another entity. Observe that the sentiment prediction performance (unlike stance prediction performance) is similar on the two sets of tweets. This shows that the two

opinion towards another entity since for some of the targets all instances in this subset are assigned to the 'neither' class, and therefore the F-score for such targets is zero on this subset.

Classifier	Atheism	Climate Change	Feminist Movemt.	Hillary Clinton	Legal. of Abortion	F-macroT	F-microT
I. Benchmarks							
a. Random	33.8	29.6	37.3	32.1	41.1	34.8	35.7
b. Majority	26.2	34.0	43.2	41.2	41.9	37.3	38.8
II. Our SVM classifier							
a. n-grams	69.7	66.9	65.3	75.9	73.2	70.2	73.3
b. a. + POS	73.3	64.2	69.9	75.1	74.5	71.4	74.4
c. a. + encodings	69.8	66.2	67.8	75.9	72.9	70.5	73.5
d. a. + sentiment	**76.9**	**72.6**	**70.9**	80.1	**80.7**	**76.4**	**78.9**
e. b. + enc. + sent.	76.3	70.6	70.5	**80.7**	79.2	75.5	78.1

Table 3.8: Sentiment Classification: F-scores obtained for each of the targets (the columns) by the benchmark systems and our classifier. Macro- and micro-averages across targets are also shown. Highest scores are shown in bold. Note 1: 'enc.' is short for encodings; 'sent.' is short for sentiment. Note 2: Even though results are shown for subsets of the test set corresponding to targets, unlike stance classification, for sentiment, we do not train a separate model for each target.

sets of tweets are not qualitatively different in how they express an opinion. However, since one set expresses an opinion about an entity other than the target of interest, detecting stance towards the target of interest from them is notably more challenging.

3.7 Stance Classification using Additional Unlabeled Tweets

Classification results can usually be improved by using more data in addition to the training set. In the sub-sections below, we explore two such approaches when used for stance classification: distant supervision and word embeddings.

3.7.1 Distant Supervision

Distant Supervision makes use of indirectly labeled (or weakly labeled) data which is mainly leveraged for supervised text classification wherein the training data is automatically generated using certain indicators present in the text. This approach is widely used in automatic relation extraction, where information about pairs of related entities can be acquired from ex-

	F-macroT		F-microT	
Classifier	To Target	To Other	To Target	To Other
I. Benchmarks				
a. Random	33.8	36.6	29.2	34.6
b. Majority	38.4	36.1	40.0	36.9
II. Our SVM classifier				
a. n-grams + sentiment	75.8	76.2	78.9	79.0

Table 3.9: Sentiment Classification: F-scores obtained on tweets with opinion towards the target and on tweets with opinion towards another entity.

ternal knowledge sources such as Freebase or Wikipedia (Craven and Kumlien, 1999; Mintz et al., 2009). Then, sentences containing both entities are considered positive examples for the corresponding relation. In sentiment and emotion analysis, weakly labeled data can be accumulated by using sentiment clues provided by the authors of the text–clues like emoticons and hashtags. For example, Go et al. (2009) extracted tweets that ended with emoticons ':)' and ':('. Next, the emoticons were removed from the tweets and the remaining portions of the tweets were labeled positive or negative depending on whether they originally had ':)' or ':(', respectively. Central to the accuracy of these sentiment labels is the idea that emoticons are often redundant to the information already present in the tweet, that is, for example, a tweet that ends with a ':)' emoticon likely conveys positive sentiment even without the emoticon. Mohammad (2012) and Kunneman et al. (2014) tested a similar hypothesis for emotions conveyed by hashtags at the end of a tweet and the rest of the tweet. Recently, distant supervision has been applied to topic classification (Husby and Barbosa, 2012; Magdy et al., 2015), named entity recognition (Ritter et al., 2011), event extraction (Reschke et al., 2014), and semantic parsing (Parikh et al., 2015).

Distant supervision

In this section, we test the validity of the hypothesis that in terms of conveying stance, stance-indicative hashtags are often redundant to the information already present in the rest of the tweet. In Section 3.7.1, we show how we compiled additional training data using stance-indicative hashtags, and used it for stance classification.

Redundancy of Stance-Indicative Hashtags

Given a target, stance-indicative (SI) hashtags can be determined manually (as we did to collect tweets). We will refer to the set we compiled as *Manual SI Hashtags*. Note that this set does

System	Accuracy
a. Random Baseline	50.0
b. Hashtag-based classification	68.3

Table 3.10: Accuracy of Favor–Against Classification on the 555 instances of the Stance Test set which originally had the manually selected stance-indicative hashtags.

not include the manually selected stance-ambiguous hashtags. Also, recall that the Manual SI Hashtags were removed from tweets prior to stance annotation.

To determine the extent to which an SI hashtag is redundant to the information already present in the tweet (in terms of conveying stance), we created a stance classification system that given a tweet-target instance from the Stance Test set, assigns to it the stance associated with the hashtag it originally had. Table 3.10 shows the accuracy of Favor–Against Classification on the 555 instances of the Stance Test set which originally had the manually selected SI hashtags. Observe that the accuracy is well above the random baseline indicating that many SI hashtags are used redundantly in tweets (in terms of conveying stance). This means that these hashtags can be used to automatically collect additional, somewhat noisy, stance-labeled training data.

Classification Experiments with Distant Supervision

If one has access to tweets labeled with stance, then one can estimate how well a hashtag can predict stance using the following score:

$$H(hashtag) = max_{stance\_label \in \{favor, against\}} \frac{freq(hashtag, stance\_label)}{freq(hashtag)} \tag{3.3}$$

where $freq(hashtag)$ is the number of tweets that have that particular hashtag; and, $freq(hashtag, stance\_label)$ is the number of tweets that have that particular hashtag and stance label. We automatically extracted stance-indicative hashtags from the Stance Training set, by considering only those hashtags that occurred at least five times and for which $H(hashtag) > 0.6$. We will refer to this set of automatically compiled stance-indicative hashtags as *Automatic SI Hashtags*. Table 3.11 lists examples.

We used both the Manual SI Hashtags and the Automatic SI Hashtags as queries to select tweets from the Stance Domain Corpus. (Recall that the Stance Domain Corpus is the large set of tweets pertaining to the five targets that was not manually labeled for stance.) We will refer to the set of tweets in the domain corpus that have the Manual SI Hashtags as the *Manual Hashtag Corpus*, and those that have the Automatic SI Hashtags as the *Automatic Hashtag*

Target	# hashtags	Favor hashtag	Against hashtag
Atheism	14	#freethinker	#prayer
Climate Change	9	#environment	-
Feminist Movement	10	#HeForShe	#WomenAgainstFeminism
Hillary Clinton	19	-	#Benghazi
Legal. Abortion	18	#WomensRights	#AllLivesMatter

Table 3.11: Examples of SI hashtags compiled automatically from the Stance Training set.

Corpus. We then assign to each of these tweets the stance label associated with the stance-indicative hashtag they contain. These noisy stance-labeled tweets can be used by a stance-classification system in two ways: (1) by including them as additional training data, (2) by capturing words that are associated with a particular stance towards the target (word–stance associations) and words that are associated with a target (word–target associations), and using these associations to generate additional features for classification.[13]

On the one hand, method 1 seems promising because it lets the classifier directly use additional training data; on the other hand, the additional training data is noisy and might have a very different class distribution than the manually labeled training and test sets. This means that the additional training data can impact the learned model disproportionately and adversely. Thus we experiment with both methods.

Table 3.12 shows the results obtained on the Stance Test set when our stance classifier is trained on various training sets. Observe that using additional training data provides performance gains for three of the five targets. However, marked improvements are observed only for 'Hillary Clinton'. It is possible, that in other test sets, the pseudo-labeled data is too noisy to be incorporated as is. Thus, we next explore incorporating this pseudo-labeled data through additional features.

The association between a term and a particular stance towards the target is calculated using pointwise mutual information (PMI) as shown below:[14]

$$PMI(w, stance\_label) = log_2 \frac{freq(w, stance\_label) * N}{freq(w) * freq(stance\_label)} \tag{3.4}$$

where $freq(w, stance\_label)$ is the number of times a term w occurs in tweets that have ***stance_label***; $freq(w)$ is the frequency of w in the corpus; $freq(stance\_label)$ is the number of tokens in

[13]Note that these word association features are similar to unigram features, except that they are pre-extracted before applying the machine learning algorithm on the training corpus.

[14]Turney (2002) and Kiritchenko et al. (2014b) used similar measures for word–sentiment associations.

Training Set	Atheism	Climate Change	Feminist Movemt.	Hillary Clinton	Legal. of Abortion	F-macroT	F-microT
a. Stance Train Set	65.2	**42.2**	57.7	60.2	**66.1**	**58.3**	**69.1**
b. a. + Manual Hashtag Corpus	62.2	**42.2**	50.5	**64.7**	62.9	56.5	66.0
c. a. + Automatic Hashtag Corpus	**65.8**	40.2	**57.8**	60.7	60.5	57.0	67.4

Table 3.12: F-scores of our supervised classifier (SVM with n-gram and target features) trained on different datasets. The highest scores for each column are shown in bold.

Features	Atheism	Climate Change	Feminist Movemt.	Hillary Clinton	Legal. of Abortion	F-macroT	F-microT
a. n-grams + target	65.2	42.2	57.7	60.2	66.1	58.3	69.1
b. a. + associations (Manual Hashtags)							
b1. word–target associations	65.6	42.7	59.9	57.6	62.8	57.7	69.0
b2. word–stance associations	63.0	42.2	58.3	**60.8**	63.5	57.6	68.4
b3. b1. + b2.	65.9	42.7	59.0	56.9	64.0	57.7	68.7
c. a. + associations (Automatic Hashtags)							
c1. word–target associations	64.5	**43.5**	58.7	55.3	**68.8**	58.1	**69.6**
c2. word–stance associations	65.1	42.4	59.1	59.8	64.3	58.1	69.2
c3. c1. + c2.	64.6	**43.5**	58.8	56.7	67.5	58.2	69.5
d. a. + associations (b1. + b2. + c1. + c2.)	**68.8**	43.3	**60.8**	56.2	64.1	**58.6**	**69.6**

Table 3.13: F-scores for our classifiers that use word–associations extracted from the domain corpus. The highest scores in each column are shown in bold.

tweets with label $stance\_label$; and N is the number of tokens in the corpus. When the system is trained on the Stance Training set, additional features are generated by taking the sum, min, and max of the associations scores for all the words in a tweet. Word–target association scores are calculated and used in a similar manner.

Table 3.13 shows the stance-classification results on the Stance Test set when using various word–association features extracted from the domain corpus. Observe that the use of word–association features leads to improvements for all targets. The improvements are particularly notable for 'Atheism', 'Feminist Movement', and 'Legalization of Abortion'. Also, the associations obtained from the Automatic Hashtag Corpus are more informative to the classifier than those from the Manual Hashtag Corpus.

Classifier	Atheism	Climate Change	Feminist Movemt.	Hillary Clinton	Legal. of Abortion	F-macroT	F-microT
a. n-grams + target	65.2	42.2	57.7	**60.2**	66.1	58.3	69.1
b. a. + embeddings	**68.3**	**43.8**	**58.4**	57.8	**66.9**	**59.0**	**70.3**

Table 3.14: Stance Classification: F-scores obtained by our classifier with additional word embedding features. The highest scores in each column are shown in bold.

3.7.2 Word Embeddings

Word embeddings are low-dimensional real-valued vectors used to represent words in the vocabulary (Bengio et al., 2001). (The 'low' dimensionality is relative to the vocabulary size, and using a few hundred dimensions is common.) A number of different language modeling techniques have been proposed to generate word embeddings, all of which require only a large corpus of text (e.g., Collobert and Weston (2008); Mnih and Hinton (2009)). Word embeddings have been successfully used as features in a number of tasks including sentiment analysis (Tang et al., 2014) and named entity recognition (Turian et al., 2010). Here we explore the use of large collections of tweets to generate word embeddings as additional features for stance classification. We investigate whether they lead to further improvements over the results obtained by the best system configuration discussed in Section 6 — SVM trained on the stance training set and using n-gram and target features.

We derive 100-dimensional word vectors using Word2Vec Skip-gram model Mikolov et al. (2013b) trained over the Domain Corpus (the window size was set to 10, and the minimum count to 2). Note that use of GloVe word embeddings pre-trained on 2 billion tweets (27B tokens) (Pennington et al., 2014) did not improve results. Given a training or test tweet, the word embedding features for the whole tweet are taken to be the component-wise averages of the word vectors for all the words appearing in the tweet.

Table 3.14 shows stance classification results obtained using these word embedding features over and above the best configuration described in Section 6. Observe that adding word embedding features improves results for all targets except 'Hillary Clinton'. Even though some teams participating in SemEval-2016 shared task on this dataset used word embeddings, their results are lower than those listed in Table 3.14. This is likely because they generated word embeddings from a generic corpus of tweets rather than tweets associated with the targets (as is the case with the domain corpus).

Overall, we observe that the three methods we tested here (adding noisy-labeled data as

new training instances, adding noisy-labeled data through association features, or generating word embeddings) affect different subsets of data differently. For example, the 'Hillary Clinton' subset of the test set benefited most from additional training data (Table 3.12) but failed to draw benefit from the embedding features. This different behavior can be attributed to many possible reasons, such as the accuracy of hashtag-based labels, the class distribution of the new data, the size of the additional corpus and the limitation of word embeddings that have very similar representation for words with opposing sentiments. Still, incorporating word embeddings seems a robust technique to improve the performance of stance detection in the presence of large unlabeled corpora.

3.8 Summary

We presented the first dataset of tweets annotated for both stance towards given targets and for the overall polarity of the message. The tweets are also annotated for whether the opinion is expressed towards the given target or towards another entity. Partitions of the stance-annotated data created as part of this project were used as a training and test sets in a recent shared task competition on stance detection that received submissions from 19 teams. We proposed a simple, but effective, stance detection system that obtained an F-score (70.3) higher than the one obtained by the more complex, best-performing system in the competition. We use a linear-kernel SVM classifier that leverages word and character n-grams as well as word-embedding features drawn from additional unlabeled data.

We presented a detailed analysis of the dataset and conducted several experiments to show the interactions between stance and sentiment. Notably, we showed that sentiment features are not as effective for stance detection as they are for sentiment prediction. Moreover, an oracle system that had access to gold sentiment and target of opinion annotations was able to predict stance with an F-score of only 59.6%. We also showed that even though humans are capable of detecting stance towards a given target from texts that express opinion towards a different target, automatic systems perform poorly on such data.

In the next chapter, the efficiency of topic modeling approaches for identifying reasons behind one's position is investigated. Later, these extracted topics are deployed as extra features in stance detection framework.

Chapter 4

Stance and the Reasons Behind it in Online News Comments

4.1 Introduction

There has been growing interest in subjectivity analysis of user generated contents in recent years. Stance classification, as the task of determining from the text whether the author of the text is in favor of or against a given target, is part of the same area of research. In the previous chapter, stance detection from Twitter data was introduced and a system based on common text classification features was proposed. In this chapter, we explore the task of detecting stance and the reasons behind it in online news comments.

Nowadays, several popular news agencies like Cable News Network (CNN) and British Broadcasting Corporation (BBC) allow their readers to express their opinions by commenting. One of the main differences between news comments and tweets is that there is no limitation on the number of characters for news comments. It further provides more opportunity for the author to not only express her overall position but also try to convince others by arguing why it is true or false. This kind of subjectivity is called argumentation (Wilson and Wiebe, 2005) and argumentation analysis is specifically focused on the reasons for author's overall position.

For news comments, we are not only interested in detecting the stance of the author towards our target of interest, but also in the reasons behind her position. Reason classification was first introduced as a separate task in Boltuzic and Šnajder (2014) in which the arguments were identified from a domain-dependent predefined list of arguments. It has been also called argument tagging. An argument tag is a controversial aspect in the domain that is abstracted by a representative phrase or sentence (Conrad et al., 2012). Here, the terms argument tagging

and reason classification refer to the same task and may be used interchangeably.

In this research, argument tagging is treated as a text categorization task in which one or more labels are selected from the predefined list of classes for each document. Classes correspond to argument tags, where an argument tag is a phrase or sentence representing a controversial aspect of a debate or the reason behind an author's position.

In this chapter, we are exploring the two tasks of reason classification and stance detection simultaneously. Given a news comment post, the target news and a predefined list of possible reasons to back up the stance towards the target, we first label the post by one, more than one, or none of those argument tags. Further, the stance of the author towards the given news is detected by leveraging the extracted argument tags as extra features in the classification framework. For example, consider the news-comment pair:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *"One in five cancers found with mammography and treated was not a threat to the woman's health and did not need treatment such as chemotherapy, surgery or radiation."*.....*What about the other FOUR in five. I'll take that mammogram. Thank you very much.*

Humans can deduce from this post that the author is likely against of the proposition abstracted in the news and the reason behind her position is that mammography can detect cancer early and save a person's life. In this case, as this argument is used mostly by authors who are against of the given target, extracting the argument and leveraging it for stance detection can improve the stance classifier performance.

The contributions of this chapter are as follows:

1. *Created a New Stance Dataset*: We created the first dataset of online news comments labeled for both stance and the reasons behind it (Section 4.2) and we make it available online [1]. We manually annotated 1,063 posts collected from various news agency websites corresponding to the news that covered a controversial study published in British Medical Journal about breast cancer screening (Miller et al., 2014). All these posts are annotated for both the stance and its intensity. Additionally, these posts are annotated for argument tags from a predefined list of tags (organized in a hierarchical tree structure).

2. *Proposed an Argument Tagging Framework*: We proposed a novel framework for argument tagging based on topic modeling (Section 4.3). Unlike other machine learning

[1]The dataset is available at http://www.site.uottawa.ca/ ∼diana/resources/stance_data_news_comments/

approaches for argument tagging which often require a large set of labeled data, the proposed framework is minimally supervised and merely a one-to-one mapping between the pre-defined argument set and the extracted topics is required. The proposed framework has comparable results with a supervised framework on our news comments dataset.

3. *Developed a Stance Detection System by Leveraging Argument Tags*: We developed a stance detection system by leveraging automatically-extracted argument tags in addition to other features that showed promising results in the previous chapter. Adding extracted argument tags to our SVM classifier with n-gram features improved classification performance from 62.90% to 64.55% in terms of average F-score (Section 4.4.2).

This chapter is mainly based on:

> Parinaz Sobhani, Diana Inkpen, and Stan Matwin. From argumentation mining to stance classification. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 67–77, Denver, CO, June 2015

4.2 A Dataset for Stance and the Reasons Behind It

Important results of health-related studies reported in the scientific medical journals are often popularized and broadcasted by media. Such media stories are often followed by online discussions in the social media. For our news comments dataset, we chose to focus on a controversial study published in the British Medical Journal (BMJ) in February 2014, about breast cancer screening (Miller et al., 2014). We choose this study as it can be an example of an important study for policy makers and, normally, before finalizing a new guideline, public's overall opinion is considered.

Subsequently, we collected a set of news articles that discussed and covered this study; then we collected their corresponding online comments. There are two Yahoo news articles[2], three CNN[3] and three New York Times articles[4]. Only root comments were kept and the rest

[2] http://news.yahoo.com/mammograms-not-reduce-breast-cancer-deaths-study-finds-001906555.html
https://news.yahoo.com/why-recent-mammography-study-deeply-flawed-op-ed-170524117.html

[3] http://www.cnn.com/2014/02/12/health/mammogram-screening-benefits/index.html
http://www.cnn.com/2014/02/19/opinion/welch-mammograms-canada/index.html
http://www.cnn.com/2014/03/18/opinion/sulik-spanier-mammograms/index.html

[4] http://www.nytimes.com/2014/02/12/health/study-adds-new-doubts-about-value-of-mammograms.html
http://www.nytimes.com/2014/02/15/opinion/why-i-never-got-a-mammogram.html
http://well.blogs.nytimes.com/2014/02/17/a-fresh-case-for-breast-self-exams/

(replies to the other comments) was discarded since they mostly contain user interactions and their opinion targets are not related to our target study. Our observation is similar to Jakic (2011) where the authors investigated nested comments and concluded that they are mostly irrelevant to the topic of the news. In the end, we collected a total number of 1,230 posts from all the sources. All the collected posts were cleaned by removing HTML tags and URL links.

4.2.1 Annotation

As mentioned earlier, our target is the medical study that was published in the BMJ journal about breast cancer screening (Miller et al., 2014). This study casts doubts about mammography screening by comparing death rate of two random groups, one that had mammography and the other that had manual exams by trained nurses. To provide sufficient background information, we gave the annotators a summary of the study and we asked them to read the original study and corresponding news.

Stance Annotation

We gave the same annotation instructions to the annotators as the ones described in chapter 3, section 3.2, for the task of determining the overall position towards our target medical study. Similarly, possible answers to the stance question are:

- *Favor*: The author explicitly or implicitly supports the target or something aligned with the target, possibly by arguing in its favor, or against someone/something other than the target from which we can infer that the author supports the target.
- *Against*: The author explicitly or implicitly opposes the target or something aligned with the target, possibly by arguing against it or in favor of someone/something other than the target from which we can infer that the author opposes the target.
- *Neither*: The overall position of the author is not inferable.

A post with 'Neither' label lacks sufficient evidence to be categorized as 'Favor' or 'Against' and may correspond to neutral, mixed, ambiguous, or irrelevant posts. Consider the example:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *It is impossible to judge another person's merit correctly unless their knowledge, talents, skills, and intellect are fully understood.*

This post is irrelevant to the main topic of the news and it seems to be an answer to another comment. There is no evidence in this post to judge the overall position of the author towards the target study.

In addition to the stance, in another question, we asked about the intensity of the position of the author towards the target study. Consequently, the annotators had five options to choose from: 'Strongly Favor', 'Favor', 'Neither', 'Against', and 'Strongly Against'. Consider the example:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *I'm so happy that this conversation is finally happening. I had a mammogram 30 years ago and said 'never again.' My intuition has always told me that these tests are dangerous. Would the people who 'caught it early have gotten the cancer in the first place had they not been exposed to this horrifically unnatural procedure? We'll never know. My opinion: DON'T GET A MAMMOGRAM.*

We can deduce from this post that the author has a strong opinion in favor of the target study and most likely will not change her opinion in the future. Similarly, in the next example, the author has a strong opinion against the target study:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *My mother saved her life thanks to an early warning through a mammogram. Nothing will make change my mind about the benefits of an early detection which is what most serious doctors recommend.*

We asked three annotators to label comments for the stance classification task. To measure the inter-annotator agreement, the average of Kappa between each pair of annotators was calculated. For the question about the overall position of the author ('Favor', 'Against', 'Neither'), Cohen's Kappa is 62%. For the question that asks about both stance and its intensity ('Strongly Favor', 'Favor', 'Neither', 'Against', and 'Strongly Against'), the labels have ordinal values so we used weighted Kappa to evaluate the quality of annotation [5]. For our 5-class stance annotation, the weighted Kappa is 54%. These numbers are in the range of reported agreement for stance annotation in similar works such as Walker et al. (2012c). For further experiments, we only kept those posts for which at least two annotators agreed on overall position and discarded the rest of posts, with less than 66% (two out of three annotators agree on the label) agreement, as they may be truly ambiguous.

[5]Fleiss' Kappa and Cohen's Kappa are mainly designed for categorical data where the degree of disagreement is not considered.

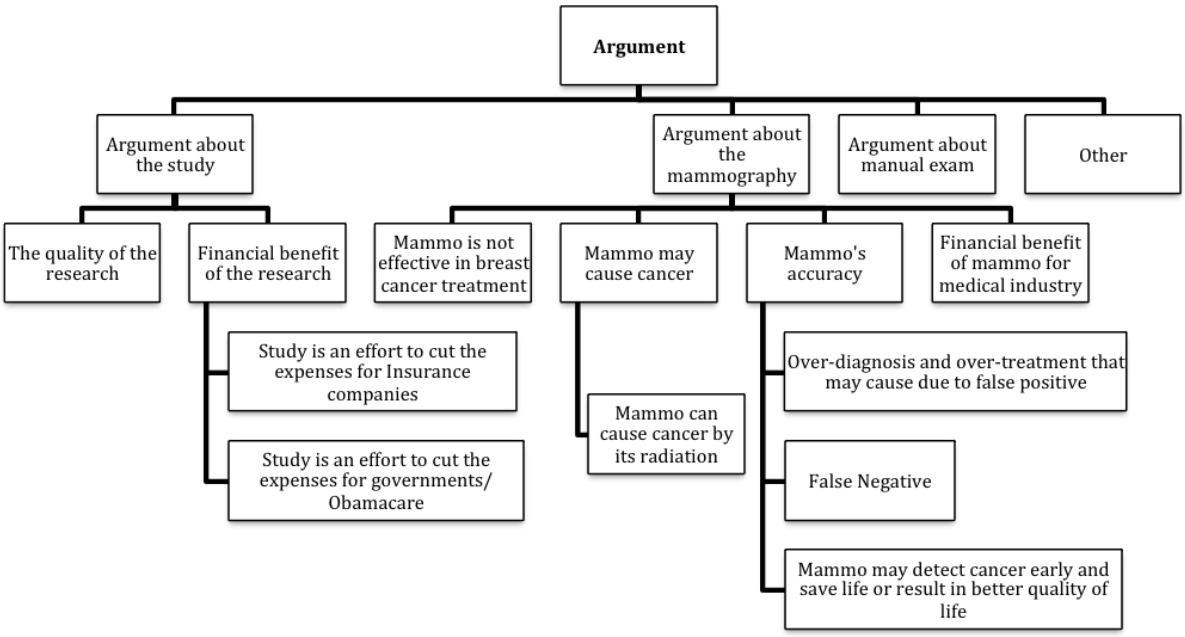

Figure 4.1: Hierarchical structure of arguments in our news comments dataset (Note: mammo is short for mammography)

Argument Annotation

For the majority of the posts, the authors back up their stances with arguments. Therefore, our second annotation task was argument tagging, in which the annotators identified which arguments were used in each comment, from a predefined list of arguments.

The given argument tags were manually extracted and organized in a hierarchical tree structure, as some of them may be related. This structure is inspired by a related field in political science called "Arguing Dimension" (Baumgartner et al., 2008). This structure is presented in figure 4.1. The annotators were instructed to give preference to the most specific arguments (the ones in the leaves of the tree) rather than more general ones. For example:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *I had my first mammogram at 40 and have one dutifully every year since. I'm now 56. Twice I've been called back in for an ultrasound, And twice I've heard the technician mumble that my breast tissue is so dense that she couldn't see anything through it. I thought to myself but didn't say out loud, "Then why do you keep giving me mammograms?" Now I will say it out loud. No more mammograms for me.*

The author of this post backs up her strong position in favor of the study by an argument about the accuracy of mammography which can not be more specified into one of the leaves labels.

Authors may have used one or more than one arguments to support their position, or sometimes no argument was mentioned. In a preliminary annotation, we found very few posts with more than two argument tags, thus we asked annotators to select at most two tags, by giving priority to those that the author put more emphasis on. Consider the example:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *Yes if it means more money in your pocket but continues to expose people to radiation unnecessarily, by all means, protest and decry the research*

We can infer that the author of the above comment is in favor of the given target study and she supports her position with two arguments: the financial benefit of unnecessary mammography for the medical industry and the radiation effect of mammography that may cause cancer.

If no argument was identified by the annotator or the argument was not in the given list, the label 'Other' should be selected. In the following example, none of the given argument tags is elaborated by the author to back up her stance:

> Target News: Study finds mammograms did not reduce breast cancer deaths
> Comment: *Study does state a reduction of death for 1 in 1000 for women over 40, 2 in 1000 for women over 50 etc. Which one of us is willing to be the sacrificial lamb who has that one death in 1000 in the world without mammograms? No one!! I'm staying with screening until I hear conclusive data suggesting mammograms cause actual (not hand waving theoretical) harm.*

We asked one of the annotators of stance detection task to also label posts by their argument tags (the predefined list of arguments are given as of figure 4.1). Additionally, to evaluate the quality of the argument annotation, a subset of this dataset (220 posts) was randomly selected and independently annotated by the second annotator. The inter-annotator agreement in terms of Cohen's Kappa, for the annotations of this randomly selected subset, was 56%. The annotations were compared without considering the hierarchical structure of the tags from figure 4.1 [6]. In Boltuzic and Šnajder (2014), a similar inter-annotation agreement was reported for argument tagging of online discussion posts.

[6]It is also possible to consider the hierarchical structure of the arguments and to calculate a weighted Kappa based on their distance in the tree.

			% of instances in Train				% of instances in Test		
Dataset	**# total**	**# train**	Favor	Against	Neither	**# test**	Favor	Against	Neither
News Comment Dataset	1,063	745	31.1	41.5	27.4	318	39.3	36.2	24.5

Table 4.1: Distribution of instances in the Online News Comments Train and Test sets for Stance

Dataset	Strongly favor	Favor	Neither	Against	Strongly against
News Comments Dataset	14.8	18.8	26.5	16.2	23.7

Table 4.2: Distribution of instances in the Online News Comments Dataset for stance and its intensity

4.2.2 Properties of the Online News Comments Dataset

Our Online News Comments Dataset has 1,063 posts. We randomly partitioned this dataset into training and test sets. 70% of the posts formed the News Comments Training set and the rest of 30% formed the News Comments Test set. Table 4.1 shows the number and distribution of instances in the News Comment Dataset for the overall stance question (3-classes) in the training and test set. Table 4.2 shows the distribution of the data for stance and its intensity (5-classes) question over the whole dataset.

Table 4.3 shows the number of instances that have been tagged for each argument. These numbers vary for different tags. For some of these tags, there are very few instances in the dataset. Following the same procedure as Hasan and Ng (2013), we replaced the low-frequency tags with "Other". In Hasan and Ng (2013), the tag is replaced if it had less than 5% total frequency. Here, we used the same threshold. After replacing low-frequency argument tags with 'Other', eight tags remained. Additionally, table 4.3 shows the distribution of stance labels per each argument tag. Note that the majority of the argument tags are biased towards one of the stance labels.

4.3 A Framework for Argument Tagging and Stance Classification

In this section, a novel framework for argument tagging and stance classification is proposed. In this framework, news comment posts are automatically clustered based on the arguments that the authors elaborated to back up their stance. These clusters are subsequently labeled

		% of instances per stance label		
Argument	**# total**	Favor	Against	Neither
Argument about the study	3	33.3	0	66.7
Argument about the mammography	3	100	0	0
Argument about manual exam	81	60.5	16.0	23.5
Mammo is not effective in breast cancer treatment	19	73.7	10.5	15.8
Financial benefit of the research	13	0	23.1	76.9
Over-diagnosis and over-treatment	99	87.9	11.1	1.0
The quality of the research	102	11.8	11.8	76.5
Cut the expenses for insurance companies	54	3.7	5.5	90.7
Cut the expenses for governments/Obamacare	77	2.6	9.1	88.3
Mammo's accuracy	17	52.9	35.3	11.8
Detect cancer and save life	261	3.1	5.7	91.2
False Negative	33	90.9	6.1	3.0
Mammo may cause cancer	10	100	0	0
Mammo can cause cancer by its radiation	76	86.8	10.5	2.6
Financial benefit of mammo for medical industry	137	74.5	24.8	0.7
Other	290	22.4	64.1	13.4

Table 4.3: Numbers of instances per argument tag and the distribution of stance labels for each argument in the Online News Comments Dataset

by considering the top keywords of each cluster returned by the adopted topic model. These labels are later leveraged in the stance classification model to determine the overall position of the author towards the target of interest.

4.3.1 Argument Tagging

As described earlier in this chapter, we formulated argument tagging as labeling each post by one or more than one tags from a given list of arguments abstracted by a phrase or sentence. Following this formulation, we can treat this problem as a multi-class, multi-output classification problem. However, to train a statistical classifier, often a large training set is required. Particularly, as the number of classes grows, we need more labeled data to have sufficient instances per class. This is the main limitation of a supervised framework for argument tagging. Instead, we propose a framework that is significantly more efficient in terms of required annotations and the human involvement is minimal.

Topic models are a group of methods in NLP that attempt to extract hidden topics from a corpus of documents. By applying a topic model on a corpus of unlabeled documents, a set of hidden topics is returned. Furthermore, each document in the corpus is represented by the mixture of these hidden topics. Our proposed framework is based on the hypothesis that arguments can be extracted as hidden topics of posts. To map these hidden topics to argument tags from the predefined list, the top keywords of each topic are considered. This is the only step in this framework that requires human supervision.

In this framework, first, a topic modeling algorithm such as Latent Dirichlet Allocation (LDA) or Non-Negative Matrix Factorization (NMF) is applied on unlabeled documents. Later, the hidden topics returned from the previous step are represented by their top keywords. These topics and a list of argument tags are given to a human annotator to map them one to one. An annotator with sufficient background about the target of interest can match topics with arguments, for any domain.

In the last step of this framework, each document is automatically labeled by an argument tag, if the probability of its matching topic is higher than a threshold for this document. These steps are depicted in figure 4.2. Here, we suggest using NMF as a topic model (we show in section 4.4 that NMF has the best performance for argument tagging compared to other topic modeling algorithms such as LDA).

In summary, in this framework, all the steps are done automatically without requiring any annotated data, except for the matching between the top keywords of each topic and the argument tags. Consequently, the suggested framework for argument tagging is considerably less

tedious and time consuming compared to annotating all posts by their tags and leveraging them for training a supervised statistical learner. For our corpus, annotating all comments took 30 hours from an annotator, while matching topics with argument tags took less than one hour.

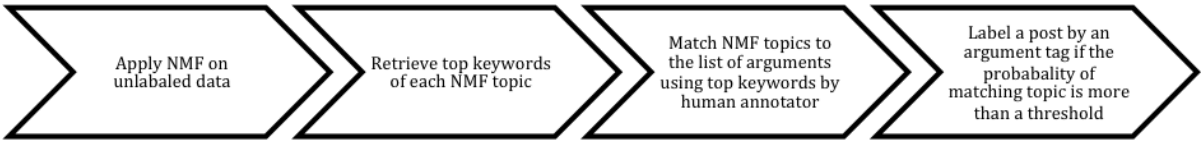

Figure 4.2: The process of tagging a post by its arguments in our proposed method for argument tagging

4.3.2 Automatically Extracted Argument Tags in Stance Classification

The correlation between stance labels and argument tags has been addressed in different studies (Boltuzic and Šnajder, 2014; Hasan and Ng, 2014). In our proposed framework, a statistical model for stance classification based on automatically extracted arguments is suggested, while in previous works stance labels were used for argument tagging. Leveraging argument tags for predicting stance labels is based on the hypothesis that an argument is often used either to support or oppose the overall position towards the target of interest. This hypothesis has been validated by analysis of our Online News Comments dataset (table 4.3) and other similar research (Boltuzic and Šnajder, 2014).

4.4 Experiments and Results

In this section, we describe the experimental setting, evaluation process, and metrics for argument tagging task. Then, we present the results obtained by our proposed framework. Later, in section 4.4.2, we present the experiments and results of the online news comments overall stance classification.

4.4.1 Argument Tagging Experiments

As described earlier in this chapter, after replacing low-frequency argument tags with ‘Other’, eight tags remained. Here, we treated argument tagging as a multi-class multi-label classifica-

tion problem where each post can be labeled by one or two of those eight labels, or none of them.

Experimental Setting

We applied different topic model algorithms on News Comments Training set. Even though these data are labeled by their argument tags, we did not use these labels and the topic model applied on unlabeled posts. The returned topics by each topic model are represented by their top ten keywords and further given to two human annotators. The annotators were asked to match these topics with the given list of eight argument tags. If the two annotators did not agree on the matching, we asked the third annotator to choose either of the assignments proposed by the other two annotators.

Each post is represented by using the Term Frequency-Inverse Document Frequency (TF-IDF) weighting scheme over its word tokens unigrams. Standard English stopwords were removed. Additionally, we removed corpus-specific stopwords by discarding tokens that appeared in more than twenty percent of the posts. Posts are first lemmatized by WordNet Lemmatizer and further tokenized by NLTK tokenizer (Bird et al., 2009). The number of topics for all topic models was set to be the number of argument tags.

We compared the results of applying our minimally-supervised framework with supervised classifier with the same TF-IDF features. As a learning algorithm, a linear-kernel multi-label Support Vector Machine (SVM) is employed using the one-versus-all training scheme. The parameters of SVM are optimized using 5-fold cross-validation on the training set.

We evaluate all the models on the News Comment Test set. As the evaluation measure, we use the average of the F1-scores (the harmonic mean of precision and recall) for the eight main classes. Note that the F1-score can be determined for all of the test instances or for each argument tag separately. We will refer to the F1-score obtained through the former method as *F-micro-across-tags* or *F-microTag* (for short). On the other hand, the F1-score obtained through the latter method, that is, by averaging the F1-score calculated for each tag separately, will be called *F-macro-across-tags* or *F-macroTag* (for short). Note that systems that perform relatively better on the more frequent argument tag classes will obtain higher F-microTag scores. On the other hand, to obtain a high F-macroTag score, a system has to perform well on all argument classes.

Results and Discussion

We now present the results obtained by our argument tagging framework on the News Comments Test set. In our proposed framework, any topic modeling algorithm can be elaborated. Here, we report experiments with Latent Dirichlet Allocation (LDA), Non-Negative Matrix Factorization (NMF) and Latent Semantic Analysis (LSA) as the most popular topic models for NLP tasks.

Table 4.4 shows the eight argument tags and their matched NMF, LSA and LDA topics represented by their corresponding top ten keywords. These mappings have been done by human annotators as described earlier. For NMF topics, the two annotators agreed on all one-to-one mappings and they found the mapping relatively easy. The topics extracted by LDA model were difficult for the annotators to map and described as vague and unclear. The two annotators agreed on 4 out of 8 mappings. For topics returned by LSA model, the two annotators agreed on 6 out of 8 mappings and while some of them were clear and easy to map; the rest were more ambiguous.

Table 4.5 presents the results of automatic argument tagging on Online News Comments Dataset. Rows I.a. to I.c. present benchmarks. I.a. shows results obtained by a random classifier (a classifier that randomly assigns one of the possible eight argument tags to each post), and Row I.b. shows results obtained by the majority classifier (a classifier that labels every instance with the majority argument tag). Observe that the F-microTag for the majority classifier is rather high while F-macroT is relatively low. Row I.c. shows results obtained by linear-kernel SVM that has been trained on News Comments Training set.

Row II.a. to II.c. shows the results obtained by our proposed framework using different topic modeling approaches. Note that the results of using NMF as the topic model are substantially better than the results of the other two topic models. The effectiveness of the NMF topics can be also observed in the extracted top keywords of the topics in table 4.4. We speculate that the reason for the better performance of NMF compared to LSA and LDA is the shortness of the comments since LDA normally works better for longer texts. Another reason may be the fact that all of these posts are about the same general topic, breast cancer screening, and LDA cannot distinguish between subtopics (different arguments). Our proposed framework using the NMF topic model has comparable performance to that of the linear SVM classifier, while it is considerably more efficient in terms of the required annotation. Note that we have an imbalanced multi-class problem and this can also be one of reasons of lower performance of the linear SVM classifier in terms of F-macroTag.

Table 4.6 provides more details about the performance of our framework using NMF topic

Argument	NMF Topic	LSA Topic	LDA Topic
1) The quality of the study	study, death, data, group, rate, survival, canadian, quality, data, result	screening, death, rate, group, treatment, canadian, article, data, poor, quality	insurance, want, company, age, test, early, treatment, screen, doctor, thing
2) Study is an effort to cut the expenses for insurance companies	insurance, company, pay, cover, sure, way, funded, benefit, expensive, money	insurance, company, money, want, care, cover, health, pay, medicine, cost	expensive, insurance, health, care, exam, save, company, money, doctor, saved
3) Study is an effort to cut the expenses for governments/Obamacare	obamacare, drop, test, past, paid, cut, obama, change, socialized, waste	obamacare, test, early, save, cover, exam, yearly, self, psa, detection	think, test, early, better, money, self, obamacare, yearly, treatment, screening
4) Mammo can cause cancer by its radiation	radiation, lumpectomy, expose, need, colonoscopy, surgery, chemo, cause, radiologist, machine	test, doctor, radiation, psa, risk, false, procedure, biopsy, unnecessary, exposure	know, radiation, mammography, cut, data, radiologist, tumor, need, surgery, medical
5) Over-diagnosis and over-treatment that may cause due to false positive	medical, false, psa, risk, needle, biopsy, screening, prostate, over, diagnosis	early, screening, treatment, test, exam, tumor, time, know, think, diagnosed	treatment, think, radiation, stage, like, make, yearly, time, article, came
6) Mammo may detect cancer early and save life or result in better quality of life	saved, stage, diagnosed, routine, early, today, discovered, mother, believe, alive	save, screen, stage, exam, history, friend, family, discovered, routine, mother	stage, radiation, saved, doctor, early, later, screening, result, want, stop
7) Financial benefit of mammo for medical industry	money, care, healthcare, medicine, people, cost, screening, preventive, administration, industry	healthcare, money, government, medicine, cost, detected, obamacare, doctor, cut, save	medicine, doctor, treatment, radiation, death, early, catching, money, save, needle
8) Argument about manual exam	exam, self, lump, tumor, manual, regular, time, malignant, trained, nurse	exam, self, tumor, detected, lump, manual, physical, annual, trained, nurse	know, people, hope, health, let, need, want, tumor, pay, radiation

Table 4.4: Topics extracted by the NMF, LSA, and LDA models represented by their top keywords

Model	F-macroTag	F-microTag
I. Benchmarks		
a. Random	13.71	14.02
b. Majority	7.06	35.51
c. Linear-SVM	43.90	50.43
II. Our Framework		
a. Cluster-LDA	21.82	28.13
b. Cluster-LSA	30.73	41.43
c. Cluster-NMF	**49.09**	**51.18**

Table 4.5: Results of automatic argument tagging on the Online News Comments Dataset

model for each argument tag. Note that better precision is achieved for argument tags that are more explicitly expressed and similar sentences are used to convey the meaning. For example, the argument "*Mammography may detect cancer early and save life or result in better quality of life*" has the best precision, as it is mostly expressed by sentences like "*Mammography saved my/my mother/ my friend life*". On the contrary, our proposed framework using NMF has better recall for those arguments referred more implicitly. Consider the example:

> Argument Tag: Study is an effort to cut the expenses for governments/Obamacare
> Comment: *Step in the direction of limited health care. You know, hope and change.*

One reason for the low precision of some of these gives argument tags, such as "*Argument about manual exam*", is that the News Comments Dataset is imbalanced (refer to table 4.3) and some of these tags have less representative instances compared to other tags.

User-Generated Content Visualization is useful for the automatic analysis of news comments in order to visualize the summary of the public opinions. Such visualizations are particularly beneficial for decision makers. In figure 4.4, we visualized the automatically-extracted argument tags from our News Comments Dataset by using our proposed framework based on NMF topic models. Similarly, in figure 4.3, the distribution of the main arguments in the corpus, based on the human annotations, are presented. In these figures, the relative importance of each of these arguments in the public opinion is visualized. Note that the distributions of argument tags in 4.3 and 4.4 are very similar. Hence, the relative importance of the given argument tags is well captured by our minimally-supervised framework. Most importantly, the same framework can be applied to any other domain without the need to label large amounts of data.

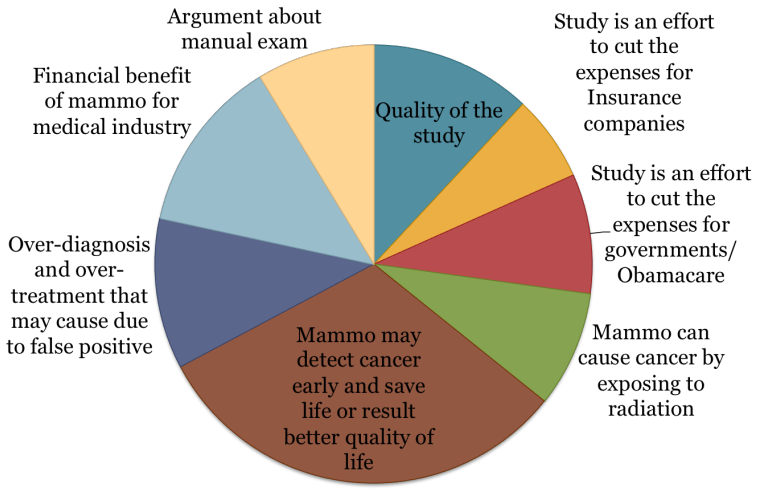

Figure 4.3: The distribution of arguments based on annotated data

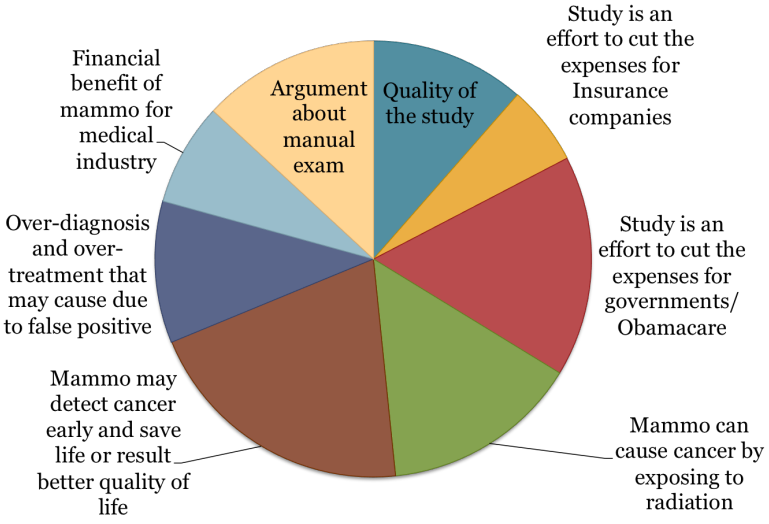

Figure 4.4: The distribution of arguments based on predicted data

Argument	Cluster-NMF		
	Precision	Recall	F1-score
The quality of the study	34	61	44
Study is an effort to cut the expenses for insurance companies	56	83	67
Study is an effort to cut the expenses for governments/Obamacare	57	24	33
Mammo can cause cancer by its radiation	33	68	44
Over-diagnosis and over-treatment	40	50	44
Mammo may detect cancer early and save life	91	38	54
Financial benefit of mammo for medical industry	44	65	52
Argument about manual exam	39	71	51

Table 4.6: The summary of the performance of proposed framework for each argument tag

4.4.2 Stance Classification Experiments

Earlier in this chapter, we proposed to use the automatically extracted argument tags for stance classification. Here, we investigate the effect of adding automatically identified argument tags as extra features for the stance classification framework.

We train a linear-kernel Support Vector Machine (SVM) classifier on the News Comments Training set. In the previous chapter, our SVM classifier with n-gram features alone showed promising results for stance classification (adding other features like sentiment lexicons features and POS tags did not improve the results significantly). Here, we use the same n-gram features (the presence or absence of contiguous sequences of 1, 2 and 3 tokens (word n-grams) and the presence or absence of contiguous sequences of 2, 3, 4, and 5 characters (character n-grams)) in our SVM classifier. We conducted 5-fold cross-validation experiments on the News Comments Training set to find the best parameters for the SVM classifier.

To investigate the importance of the predicted argument tags for stance classification, we add automatically extracted argument tags as eight extra features to our word n-grams and character n-grams. These features show the presence or absence of each of the eight main argument tags presented in table 4.4.

We evaluate the learned models on the News Comments Test set. As the evaluation measure, we use the average of the F1-scores (the harmonic mean of precision and recall) for stance classification (for the two classes, 'Favor' and 'Against'):

$$F_{avg} = \frac{F_{Favor} + F_{Against}}{2} \tag{4.1}$$

For stance and its intensity classification (the five classes, 'Strongly favor', 'Favor',

	Stance Classification	Stance and its Intensity Classification
Classifier	**F-avg**	**F-avg**
I. Benchmarks		
a. Random	35.01	18.62
b. Majority	26.56	0
II. Our SVM classifier		
a. n-grams	62.90	32.29
b. n-grams + arguments	**64.55**	**35.93**

Table 4.7: Results of stance classification (3-classes) and stance and its intensity (5-classes) on our News Comments Dataset

'Neither', 'Against', and 'Strongly against'):

$$F_{avg} = \frac{F_{StronglyFavor} + F_{Favor} + F_{Against} + F_{StronglyAgainst}}{4} \tag{4.2}$$

Table 4.7 presents the F_{avg} obtained by benchmark systems and our classifiers for stance classification (3-class problem) and stance and its intensity classification (5-class problem). Rows I.a. and I.b. show results obtained by a random classifier and majority classifier respectively. The reason for low F-scores obtained by the majority classifier is that the distribution of the training and test sets in our News Comments Dataset are slightly different, and for stance classification, the majority class is different in these two sets. In the case of stance and its intensity, the majority class is "Neither"; consequently, F_{avg} equals to zero.

Rows II.a. and II.b. show that adding the predicted argument tags to the n-gram features improves the performance of the linear-kernel SVM substantially for both stance and its intensity classification. The predicted argument tags help the stance classification because the majority of these arguments have been leveraged mainly to back up the authors' stances, either in favor or against the target of interest 4.3.

4.5 Summary

Stance classification and argumentation mining were recently introduced as important tasks in opinion mining. There has been a growing interest in these fields, as they can be advantageous particularly for decision making. In this chapter, a novel framework for argument tagging was proposed. In our approach, news comments were clustered based on their topics. These clusters were subsequently labeled by considering the top keywords of each cluster.

The main advantage of the proposed framework is its significant efficiency in the annotation. Most of the previous works required a large set of annotated data for training supervised classifiers; and the annotation process is tedious and time-consuming, while in our approach there is no need for labeled training data for the argument tagging task. The annotation needed for the argument detection task is minimal: we only need to map the automatically-detected topics to the arguments. This mapping can be easily done for new domains.

Experiments on our News Comments Dataset demonstrate the efficiency of using topic modeling for argument tagging. We show that using Non-Negative Matrix Factorization instead of other topic models such as Latent Dirichlet Allocation achieves better results for argument tagging, close to the results of a supervised classifier. Furthermore, we conducted experiments to explore the advantages of adding automatically extracted argument tags as extra features for stance classification. Our linear-kernel SVM classifier with n-gram features improved substantially by adding predicted argument tags to its set of features.

Additionally, We presented the first manually-annotated News Comments dataset for both stance classification and argument tagging. By making this dataset available, more work on joint learning of stance and the reasons behind it is encouraged.

In the next chapter, we explore deep neural networks for multi-target stance detection. We jointly model the subjectivity expressed towards multiple targets by adopting an attention-based encoder-decoder framework.

Chapter 5

Multi-Target Stance Classification

5.1 Introduction

Detecting subjectivity, for example, sentiments or stances, expressed towards different targets has a wide range of utilities. Previous work often treats each target independently, ignoring the potential dependency that exists among the targets, the corresponding subjectivities, and other hidden factors associated. For example, in an electoral social-media message, the stance expressed towards one candidate could be highly correlated with that towards another entity, and such dependency exists widely in many other domains, including product reviews. In its difficult presence, subjectivity correlation could be associated with hidden factors such as topics under concern (e.g., two political candidates are not necessarily against each other on all topics), among others.

In this chapter, one of our main goals is to provide a benchmark dataset to jointly learn subjectivities corresponding to related targets. We created a dataset of 4,455 tweets manually annotated for stance towards more than one target simultaneously. We will refer to this data as the Multi-Target Stance Dataset.

As mentioned in previous chapters, stance detection is the task of automatically determining from the text whether the author of the text is in favor of, against, or neutral towards a proposition or target. The target may be a person, organization, government policy, movement, product, etc. Over the last decade, there has been active research in modeling stance (Thomas et al., 2006; Somasundaran and Wiebe, 2009a; Anand et al., 2011; Walker et al., 2012b; Hasan and Ng, 2013; Sobhani et al., 2016). However, all of these previous works treat each target independently, ignoring the potential dependency existing among related targets.

In this chapter, we further investigate the problem of jointly predicting the stance expressed

towards multiple targets. We specifically explore this problem in a social media text, the Twitter text, where people often express stance either implicitly or explicitly. The task is formulated as follows: given a social media post and k related target entities, automatic natural language systems must jointly learn the overall position of the author towards the given targets where one prediction has a potential effect on the other ones. Formally, given a set of social media posts D related to the k target entities $T_1, ..., T_k$ our goal is to determine the value of mapping $s : T_1 \times ... \times T_k \times D \rightarrow \{favor, against, neither\}^k$ for each post $d \in D$. Our approach to solve this task is by learning the mapping using machine learning techniques. For example, consider the tweet and target pair:

Targets: Hillary Clinton & Donald Trump

Tweet: *It's true, #DonaldTrump is brutally honest sometimes, but I'd rather face a roaring lion, than a quiet scorpion. #Hillary #Election2016*

Humans can deduce from the tweet that the tweeter is likely in favor of Donald Trump and against Hillary Clinton.

This task is different from other tasks in the literature. Most close to ours is the work presented in Deng and Wiebe (2015a) where sentiment towards different entities and events is jointly modeled using a rule-based probabilistic soft logic approach. In another similar work, Deng et al. (2014) suggested an unsupervised framework to detect implicit sentiment by inference over explicit sentiments and events that positively or negatively affect the theme. They also made their dataset MPQA 3.0 (Deng and Wiebe, 2015b) available, However, their dataset is relatively small (it contains 70 documents–news articles and other text documents) and has a potentially infinite number of targets (target sets depend on the context), which makes it hard to train a system.

Instead, we provide a reasonably-large training data and the benchmark evaluation here. What is critical in order to get enough training data is to focus on subjectivity towards targets that users care more about (e.g., Hillary Clinton vs. Donald Trump). Moreover, we make available a large amount of unlabeled data that provides more possibilities for learning world-knowledge about the relationship between entities and facilitates inference about the overall position of the author towards different entities.

We propose a framework that leverages deep neural models to jointly learn the subjectivity towards more than one target entity given the text of a tweet. We treat the task as sequence-to-sequence learning, where the entire text of the tweet is mapped to a vector at the encoder side using bi-directional recurrent neural networks (RNN). On the decoder side, RNNs conditioned

on the input vectors generate stance labels towards the related entities. By using an attention-based network, the model can focus on different parts of the tweet text to generate each stance label. Because stance labels are also generated conditionally dependent on the previously generated labels towards other entities, the model removes the independence assumption between different targets labels and specifically focuses on the dependencies.
The contributions of this chapter are as follows:

1. *Created a multi-target stance dataset*: We created the first dataset of tweets labeled for more than one target per post for stance. More than 4400 tweets are annotated for whether one can deduce favorable or unfavorable stance towards two targets simultaneously. We collected tweets related to 2016 US elections, with three target pairs for our Multi-Target Stance Dataset: 'Donald Trump and Hillary Clinton', 'Donald Trump and Ted Cruz' and 'Hillary Clinton and Bernie Sanders'. The construction of the dataset is described in:

    - Parinaz Sobhani, Xiaodan Zhu, and Diana Inkpen. A dataset for multi-target stance detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, Valencia, Spain, April 2017

2. *Explored the problem of stance detection towards the related target of interests:* We explored the problem of jointly detecting subjectivity expressed towards concerned targets while previous work often treated each target independently, ignoring the potential dependency among targets. We experimentally showed the efficiency of jointly learning dependent subjectivities compared to independently predicting subjectivities towards individual targets.

3. *Developed a deep neural system for multi-target stance detection:* We developed a system for multi-target stance detection by leveraging neural models, particularly attention-based encoder-decoder framework. We showed that the proposed model is more effective in jointly modeling the overall position towards two related targets, compared to other popular frameworks for joint learning, such as cascading classification and multi-task learning.

5.2 Dataset

We collected tweets related to the 2016 US election. We selected four presidential candidates: 'Donald Trump', 'Hillary Clinton', 'Ted Cruz', and 'Bernie Sanders' as our targets of interest

and identified small set of hashtags (which are not stance-indicative) related to these targets[1]. We polled the Twitter API to collect more than eleven millions of tweets containing any of these query hashtags. For approximately 25% of the tweets, the hashtag of interest appeared at the end. Hashtags at the end of the tweets may not have any contribution to the meaning of the tweets; it means that the targets of opinion may not be the same as the targets of interest and therefore inference is required. This is one of the main differences between our task and aspect-based sentiment analysis. Here is an example from our dataset:

> Tweet: *Given a choice to kill 100 ISIS or 100 white American men, leftist scum would choose the latter. #UniteBlue #nomorerefugees #Bernie #Hillary*

In this example, none of the targets of interest, 'Hillary Clinton' or 'Bernie Sanders' are mentioned explicitly in the tweet text except by the hashtags at the end of the tweet, but humans can infer that the tweeter is likely against both of them.

5.2.1 Data Annotation

We selected three target pairs for our Multi-Target Stance Dataset: Donald Trump and Hillary Clinton, Donald Trump and Ted Cruz, Hillary Clinton and Bernie Sanders. Further, we filtered the collected tweets by removing short tweets, retweets and those having a URL. We also discarded tweets that do not include at least two hashtags, one for each of the targets of interest. For each of the three selected target pairs, we randomly sampled 2,000 tweets. These tweets were annotated through CrowdFlower[2]. We asked the annotators two questions, one for the stance towards each of the presidential candidates in the target pair of interest. For stance annotation, the same annotation instructions were used as described in section 3.2.2.

We used CrowdFlower's gold annotations scheme for quality control, wherein about 10% of the data was annotated internally (by the authors). During crowd annotation, these gold questions were interspersed with other questions, and the annotator was not aware which is which. However, if she got a gold question wrong, she was immediately notified of it. If the accuracy of the annotations on the gold questions falls below 70%, the annotator was refused further annotation. This served as a mechanism to avoid malicious annotations and as a guide to the annotators.

Each tweet was annotated by at least eight annotators. To aggregate stance annotation information from multiple annotators for an instance rather than opting for a simple majority,

[1]Our hashtags list includes: #DonaldTrump, #Trumpt, #Trump2016, #TedCruz, #Cruz, #Cruz2016, #TedCruz2016, #HillaryClinton, #Hillary, #Hillary2016, #BernieSanders, #Bernie, #Bernie2016

[2]http://www.crowdflower.com

Target Pair	# total	# train	# dev	# test
Clinton-Sanders	1366	957	137	272
Clinton-Trump	1722	1240	177	355
Cruz-Trump	1317	922	132	263
Total	4455	3119	446	890

Table 5.1: Distribution of instances in the Train, Development and Test sets for different target pairs in the Multi-Target Stance Dataset

the instances with less than 50% agreement on any of the candidates in the target pairs were discarded. We refer to this dataset as the Multi-Target Stance Dataset and we make it available online [3]. The inter-annotator agreement on this dataset is 79.74%. We kept the rest of the tweets that were not used in the annotation process as unlabeled data, which can be used to obtain additional information about stance and relations between relevant entities.

5.2.2 Properties of the Multi-Target Stance Dataset

We partitioned the Multi-Target Stance Dataset into training, development, and test sets based on the timestamps of the tweets. All annotated tweets were ordered by their timestamps, and the first 70% of the tweets formed the training set, the next 10% as development set the last 20% formed the test set. Table 5.1 shows the number of instances in the training, development, and test sets over different target pairs in our Multi-Target Stance Dataset.

Having different US presidential candidates as targets of interest does not imply that the tweeter may necessarily have opposing positions towards them. There are several cases where authors have favorable stances towards both, or similarly, opposing positions towards both of them. In our dataset, approximately 20% of the tweeters have the same position towards both entities, 50% of tweeters have opposing positions towards the given targets, and for 17% of the data, the positions towards none of the targets is inferable. The examples below show tweets that have the same position towards two candidates; the first one favorable towards both, and the second one against both of them:

> Targets: Hillary Clinton & Bernie Sanders
>
> Tweet: *Bottom line, I'll take either one as president, they'll have their strengths and weaknesses, but should do a good job. #Hillary #Bernie*

[3]The dataset is available at http://www.site.uottawa.ca/ ∼diana/resources/stance_data/

Opinion towards		Clinton		
		favor	against	neither
Sanders	favor	7.5	33.9	3.7
	against	12.6	12.0	3.8
	neither	2.3	5.6	18.6

Table 5.2: Confusion between stance labels for Hillary Clinton-Bernie Sanders target pair

Opinion towards		Clinton		
		favor	against	neither
Trump	favor	0.5	52.3	1.2
	against	14.0	9.0	3.5
	neither	0.3	3.9	15.2

Table 5.3: Confusion between stance labels for Donald Trump-Hillary Clinton target pair

Targets: Donald Trump & Hillary Clinton

Tweet: *Looking at the List of PC's for 2016 is like looking at the McDonalds Menu. You just know that shit is bad for you. #Trump2016 #Hillary2016*

To illustrate more details about the correlation between subjectivities towards targets of interest, the confusion between stance labels of different target pairs in the Multi-Target Stance Dataset are depicted in tables 5.2, 5.3 and 5.4. We note that the numbers vary between target pairs.

5.3 Multi-Target Stance Classification

In this section, we propose a framework that leverages recurrent neural models to capture the potentially complicated interaction between subjectivities expressed towards multiple targets. We experimentally show that the attention-based encoder-decoder framework is more effective in jointly modeling the overall position towards two related targets, compared to independent predictions of positions and other popular frameworks for joint learning, such as cascading classification and multi-task learning.

Opinion towards		Cruz		
		favor	against	neither
Trump	favor	18.7	22.5	2.8
	against	10.3	17.4	4.8
	neither	3.3	2.3	18.0

Table 5.4: Confusion between stance labels for Ted Cruz-Donald Trump target pair

5.3.1 Window-Based Classification

One popular approach to detect subjectivity towards different targets, as is used in aspect-based sentiment classification (Brychcın et al., 2014), is to consider a context window of size n in both directions around the target terms and to extract features for that target's classifier based on its context. This approach is based on the assumption that the words outside the context window do not have an influence on the target. We will first include such a baseline for our task.

5.3.2 Cascading Classifiers

To capture dependencies between stance labels of related targets, one possibility is to use the predicted class towards one target as an extra feature in other targets' models. This framework is based on cascade classification, where several classifiers of related tasks are combined to improve the overall system performance (Heitz et al., 2009). we adopted this framework for multi-target stance classification.

Cascade classification framework consists of classifiers conditionally dependent on the output of other learners in addition to the independent classifiers. We used this framework for multi-target stance classification by starting from an independent classifier to predict stance towards the first target based on the text representation and exploit its prediction as an extra feature for other classifiers. One of the main challenges in this framework is the order of predicting targets. Here, we tried all possible orders and trained a model for them; subsequently, at the test time, we have predictions of different cascade classifiers, and the one with the highest confidence is selected.

The major restriction of this framework is that the classification algorithm should have a mechanism to add new features based on other learners outputs. Most of the machine learning algorithms for text classification that rely on hand-crafted features extracted from text to represent it provide such mechanism. But, for the state-of-the-art deep neural models, where

the feature vectors for the text representation are learned with the classification model during training, adding new features to the model is not trivial.

5.3.3 Recurrent Neural Networks for Multi-Target Stance Detection

Recurrent Neural Networks (RNN) (Elman, 1990) are most popular models for sequential data and have shown promising results in various NLP task. Basically, they are extensions of feed-forward neural models that can handle variable-length inputs. They are particularly beneficial for sequential data because unlike a feed-forward network, they share their hidden states across time and keep track of the information processed as a memory. The sequential history is summarized in a hidden vector. A standard RNN iterates over the input sequence and computes the hidden state s_t at each step as follows:

$$s_t = f(Ux_t, Ws_{t-1}) \tag{5.1}$$

where f is a non-linear function such as sigmoid and x_t is the current input and s_{t-1} is the previous hidden state. The first hidden state is normally initialized to all zeros.

While Vanilla RNN might suffer from the decaying of the gradient, or less frequently, blowing-up of gradient problems, different architecture of RNN such as Long Short-Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) and Gated Recurrent Unit (GRU) (Cho et al., 2014a) can capture long-term dependencies by having different gating mechanisms. LSTM replaces the hidden vectors of a recurrent neural network with memory blocks which are equipped with gates; it can in principle keep long-term memory by training proper gating weights. Similarly, GRU has gating units that control the flow of information into the unit, but, without having separate memory cells (Chung et al., 2014). Even though there are differences between these two types of gated recurrent neural networks, various works (Bahdanau et al., 2014; Chung et al., 2014) reported comparable performance for them.

RNN later extended as bi-directional RNN (Irsoy and Cardie, 2014b; Schuster and Paliwal, 1997) as the output at each time step not only depends on previous elements in the sequence but also might depend on the next elements in the sequence. A bi-directional RNN consists of two RNNs, the one that processes the input in its original order and the one that processes reversed input sequence. Finally, the output is the function of hidden states of both RNNs.

Multi-Task Bi-directional RNN

Multi-task learning has become extremely popular within natural language processing and machine learning over the last few years. The main benefit of multi-task learning is improving generalization by taking advantage of the inductive bias in training signals of related tasks. Additionally, it can help the classification model in several other ways. It can increase the number of training data particularly when there is a limited number of training data in each task. While Inducing the model only from the sparse single task data may lead to overfitting to random noise in the data, by leveraging auxiliary data from other tasks, the model can abstract away from the noise. Multi-task learning can also be motivated from a representation learning perspective, as a way to use auxiliary tasks to induce representations that may be beneficial for the target task. Finally, we can cast multi-task learning as a regularizer for example by reducing the Rademacher complexity in multi-task architectures over the single-task model.

We adopted a multi-task learning (MTL) architecture (Liu et al., 2015) based on bi-directional RNN, as described earlier in this section. In this framework, MTL can be seen as a way of regularizing model induction by sharing the embedding and bi-directional RNN layers and having a distinct softmax layer on top, for each target, i.e., each target's stance detection is treated as a different task and is associated with an independent classification function, but they share the hidden layers.

We assume K different training sets corresponding to K different targets, $D_1, \cdots, D_K$, where each D_k corresponds to pairs of input sequence and the output $(x_{1:m}, y)$, where $x_i \in V$, $y \in L$. The input vocabulary V and output vocabulary (stance labels) L are shared across tasks. At each step in the training process, we choose a random task k, followed by a random training batch from D_k. We use the model to predict the labels $\hat{y}^i$, measure the loss with respect to the true labels y^i, and update the model parameters. The objective function is to minimize the sum of the losses of the K models on their respective training datasets.

Sequence-to-Sequence Model to Capture Dependencies in Output Space

Encoder-decoder sequence-to-sequence models (Sutskever et al., 2014; Cho et al., 2014b) were originally used for machine translation, where a bi-directional RNN is trained to learn the representation for the source language and the other RNN generates the translation in the target language. Later, this approach was adopted for other tasks with variable input and output sequence lengths, such as speech recognition (Hannun et al., 2014) and question answering (Hermann et al., 2015).

We propose to use the attention-based encoder-decoder for multi-target stance classifica-

tion. Specifically, we will regard the given tweet as the input, and the model is trained to generate the stance labels for targets. This model can naturally capture the dependencies existing among the target stance labels when searching the best labels sequence, based on automatically learned input features. The attention mechanism has the potential of dynamically focusing on different words of the input text to generate stance labels for each target of interest. As such, the attention-based encoder-decoder is expected to have the strengths of both the window-based classification, by dynamically customizing the feature vector to predict each target stance label, and the cascading classification, by conditioning each label generation on the other labels without inheriting the limitations of these models. The model automatically learns the features and the regions of the input that should be paid attention to.

Given an input sequence $x = x_1, x_2, ..., x_m$ and an output sequence $y = y_1, y_2, ..., y_k$, a sequence-to-sequence approach models the conditional probability of $p(y|x)$. This model consists of two RNNs: one to model the input sequence (encoder) and the other one to generate the output sequence (decoder) where the last hidden state of the encoder is used to initialize the hidden state of the decoder.

To overcome the limitation of sequence-to-sequence models, where the last state of the encoder is supposed to summarize all the information in the input sequence, the attention-based model suggested in Bahdanau et al. (2014). Figure 5.1 depicts the attention-based decoder-encoder model in our multi-target stance detection task. Attention mechanism enables the model to automatically searches for more relevant parts of the source to generate the next target word. In this model, each output y_t is generated at the decoder side as a function of the previous generated output y_{t-1}, the previous hidden state s_t and a context vector c_t.

$$p(y_t|y_{t-1}, ..., y_1, x) = g(y_{t-1}, s_t, c_t) \tag{5.2}$$

where g is a softmax function, and s_t is the hidden state of the decoder at time t and is calculated as follows:

$$s_t = f(y_{t-1}, s_{t-1}, c_t) \tag{5.3}$$

where c_t is the weighted summation of the input sequence:

$$c_t = \sum_{i=1}^{n} \alpha_{ti} h_i \tag{5.4}$$

and α is calculated based on the similarity between the hidden state of input at position t (h_t) and the hidden state of the output at position t (s_t).

$$\alpha_{ti} = sim(s_{t-1}, h_i) \tag{5.5}$$

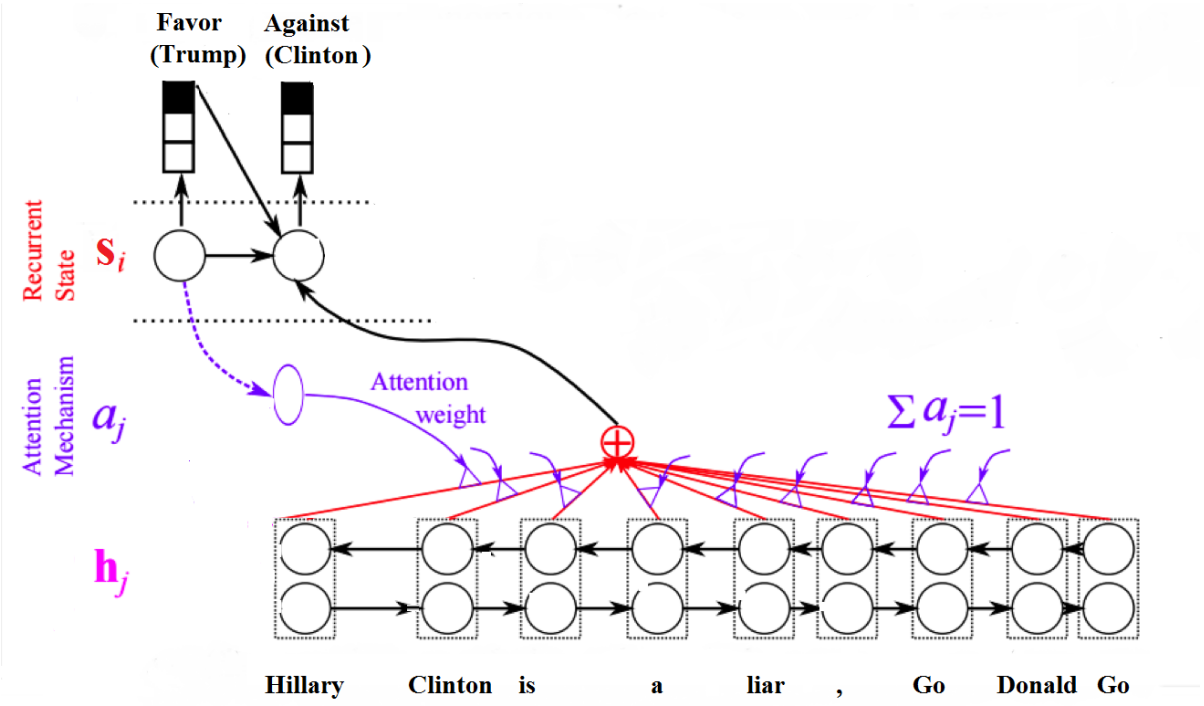

Figure 5.1: The attention-based encoder-decoder framework deployed for multi-target stance detection (This figure is largely adopted from Kyunghyun Cho Deep's Natural Language Understanding slides `http://videolectures.net/deeplearning2016_cho_language_understanding/`).

We adopted an attention-based sequence-to-sequence model for multi-target stance classification for several reasons. First of all, to have a general framework that can handle variable numbers of targets of interest. The other main advantage of this model is that it can capture dependencies in the output space, since generating the label towards each target is based on the previous targets' labels, as well as on the linear combination of the input sequence. Another advantage of this approach is the attention mechanism which enables the model to dynamically focus on different words of the input text to generate stance labels for each target of interest. Particularly, the attention-based sequence-to-sequence model described above has the strengths of both the window-based classification, by dynamically customizing the feature vector to predict each target stance label, and the cascading classifier, by conditioning each label generation on the other labels generated before, without inheriting the limitations of the two models. It is also similar to the multi-task framework by sharing the tweet representation for different targets, while the sequence-to-sequence model is more powerful than multi-task learning by having the capability of capturing the dependencies between targets and attention mechanism.

5.4 Experiments

We evaluate the effectiveness of our models on the Multi-Target Stance Dataset described earlier, where two stance labels are predicted for each tweet. Note that all the models can be easily extended to predict more than two labels as well. For all methods, the tweets were tokenized with the CMU Twitter NLP tool (Gimpel et al., 2011). All the models we proposed here, were implemented in Python. To implement independent LSTMs and Multi-task LSTM we used Keras deep learning library (Chollet, 2015) and to implement attention-based encoder-decoder model, we deployed Theano library (Theano Development Team, 2016).

Unlike previous works for sentiment classification that learn a single model from all of the training data, we trained a different network per target, for multi-task learning a joint model per target pair and for sequence-to-sequence an encoder-decoder model per target pair. The reason is that for sentiment classification, only one labeled is assigned to each tweet, while here a label is assigned for each target, and the targets of interest are different. Moreover, the words and concepts used in tweets corresponding to different targets might not generalize across the targets.

For encoder-decoder attention-based models, we trained two models per target pair, for two different orders of the targets (For example for Clinton-Trump pair, we trained a model that

first predicts the stance towards Clinton and then the stance towards Trump and the other model that predicts stance labels in the reverse order of the targets). At test time, for each tweet, each model returns stance labels together with a score, and the prediction from the model with the higher confidence is selected.

5.4.1 Training Details of Different RNN Models

We applied different RNN architectures to our Multi-Target Stance Dataset. For all experiments, we followed Bahdanau et al. (2014); Luong et al. (2015) to train our models using the minibatch stochastic gradient descent (SGD) algorithm with adaptive learning rate (Adadelta (Zeiler, 2012)). The minibatch size is set to 32. RNN units have 128 cells and are initialized randomly. The RNN layer is placed on top of an embedding layer with 100 dimensions where word vectors are pretrained.

Pretraining: Deep neural models normally require large training sets to learn all their networks parameters. In our task, for each target pair, we have approximately 1,000 training tweets, which is relatively small. As a remedy and to exploit the large amount of related unlabeled tweets (11,873,771 tweets) that we collected in the same time period, we initialized our word representations (embedding layer) with 100-dimensional word vectors trained on the unlabeled tweets pertaining to the four targets: 'Hillary Clinton', 'Bernie Sanders', 'Donald Trump' and 'Ted Cruz'. As a training algorithm, we employed the Word2Vec Skip-gram model (Mikolov et al., 2013b).

We limit the input vocabulary list V for each model to the words that appeared at least two times in the tweets from the training set. The rest of the tokens that are not in this vocabulary list are replaced by the universal token $< unk >$. For sequence-to-sequence model, the vocabulary list for the target side is limited to four tokens: "Favor", "Against", "Neither" and the special token for the end of the target side $< EOS >$, as we need to predict the stance labels.

All models aim to optimize the training objective which is:

$$J_t = \sum_{(x,y)\in D} -logp(y|x) \tag{5.6}$$

where D is our training set containing tweets and their corresponding label/labeles. The model with the best performance on the development set is selected at the end of the training.

5.4.2 Evaluation Metric

As the evaluation measure for each target, we use the average of the F1-scores (the harmonic mean of precision and recall) for the two main classes, Favor and Against:

$$F_{avg} = \frac{F_{favor} + F_{against}}{2}$$

Note that this measure does not give any credit for correctly classifying “Neither" instances. Nevertheless, the model has to predict all three classes (favor, against, and neither) to avoid being penalized for misclassifying “Neither" instances as in favor or against. A similar metric was used in the past for sentiment analysis—SemEval 2013 Task 2 (Wilson et al., 2013a) and for stance detection—SemEval 2016 Task 4 (Mohammad et al., 2016c). For multi-target (in our dataset, target pairs) the average of F_{avg} over all the targets is calculated. To report a single number for all three target pairs, we take the average of three values returned for each target pair and we refer to it as macro-averaged F-score. All the models are evaluated on the test sets.

5.4.3 Results and Discussion

Table 5.5 presents the macro-averaged F-scores of different models on the Multi-Target Stance dataset. Row i. shows the result obtained by a random classifier (a classifier that randomly assigns a stance class to each instance), and row ii. shows the result obtained by the majority classifier (a classifier that simply labels every instance with the majority class per target).

When we have multiple targets to predict overall positions towards them, one possibility is to have a single learner per target that are independently trained. Row a. shows the result of having two independent LSTMs. Row b. shows the result of having two independent Support Vector Machine (SVM) classifiers. We used a linear kernel SVM applied to unigram features. SVM is a state-of-the-art learning algorithm proved to be effective on various text categorization tasks and robust on large feature spaces. We used the implementation provided in the Scikit-learn Machine Learning library Pedregosa et al. (2011). The parameters of the SVM models are tuned using development datasets. For the majority of targets, SVM model has a better performance compared to LSTM model, it might be due to the limited number of training instances; as for Clinton-Trump pairs that have approximately 25% more training data, the performance of two models are comparable in terms of average f1-score.

Row c. is the result of applying Window-based SVM on our Muti-Target Stance Dataset. Because we collected our data based on hashtags related to the targets, those hashtags can be considered as target terms and we place a context window around them. We used the development set to find the best value for the window size. The main limitation of this approach on this

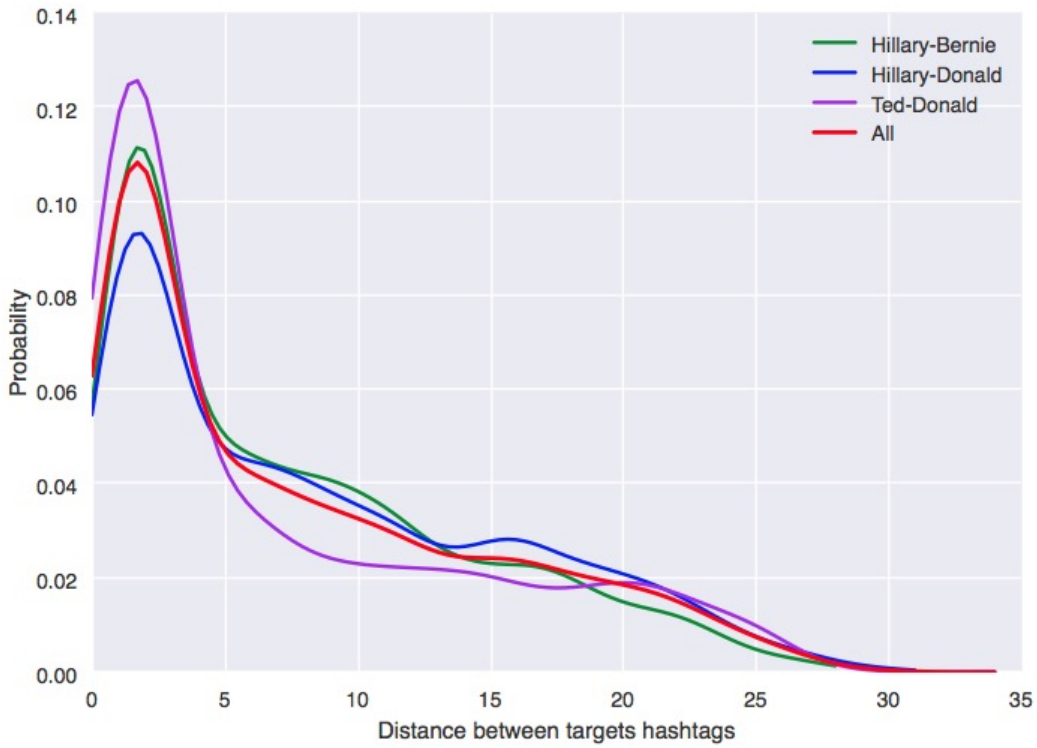

Figure 5.2: The distribution of distance between two target hashtags in the dataset for different pairs and in overall (the distance between two target hashtags is in terms of number of words)

dataset is that for the majority of the tweets, the contexts windows have significant overlaps, as the two hashtags appeared in the close vicinity of each other. For approximately 10% of tweets, both hashtags appeared at the end of the tweets. Figure 5.2 shows the distribution of the distance between two target hashtags in the dataset, for different pairs and overall.

Row d. presents the results of the Cascading SVMs and it shows improvement over the baseline of independent SVMs. As explained earlier, cascading is a popular framework for jointly predicting several outputs.

Another possibility when there is more than one output to predict is to combine all the outputs and train a single model. For our task of predicting stance towards a target pair, where each can take one of the three possible labels: “Favor", “Against" and “Neither", combining the two prediction results in a 9-class learning problem. Row A. shows the result of this classifier. The main limitation of combining outputs is that the number of classes can grow substantially while there is a fixed number of labeled instances which results in a drop in performance,

Classifier	F-macro
Benchmarks	
i. Random	34.26
ii. Majority	32.11
One Classifier per Target	
a. Independent LSTMs	50.40
b. Independent SVMs	51.37
c. Window-based SVMs	48.32
d. Cascading SVMs	52.05
Single Model	
A. 9-Class SVM	50.63
B. Multi-Task LSTM	51.62
C. Seq2Seq	**54.81**

Table 5.5: Macro-averaged F-scores of different models on the Multi-Target Stance dataset

according to the Probability Approximation Correct learning framework (Kearns and Vazirani, 1994). Another issue is that some of the classes might not have enough representative instances and this can lead to a highly imbalanced classification problem.

Row B. is the result of multi-task LSTM in average over all targets where stance labels of each target pairs are learned jointly by sharing the embedding and LSTM layers. Finally, row C. shows the results of applying the attention-based encoder-decoder deep neural model on Multi-Target Stance Dataset. This model has both the advantages of windows-based and cascading classification, and it has the best performance in terms of macro-average F-score, compared to all other models and baselines. Comparing its result to multi-task LSTM shows the importance of attention mechanism and capturing the dependencies in the output space.

Table 5.6 shows more details for the performance of different models on each target pair. By applying paired t-test on these results, we concluded that, for a 95% confidence interval, the differences between sequence-to-sequence model and all other models are significant. It can be observed that for Clinton-Trump and Cruz-Donald joint stance prediction, our bi-directional encoder-decoder (seq2seq), substantially outperforms other models. While for the Clinton-Sanders pair, the seq2seq model has comparable performance with independently-trained SVMs. Particularly for the 'Clinton' target in 'Clinton-Sanders pair, none of the models could improve over the independently-trained SVMs. A possible explanation for this is that most of the tweets for this target are classified as 'Against', hence, the training data is imbalanced, and conditioning the classifier on the prediction of the stance towards'Bernie Sanders' made the situation worse, by enforcing the classifier to predict more 'Against' labels.

	Clinton-Sanders			Clinton-Trump			Cruz-Trump		
Classifier	Clinton	Sanders	Avg	Clinton	Trump	Avg	Cruz	Trump	Avg
Random Classifier	30.82	34.11	32.47	36.01	29.62	32.82	37.63	37.36	37.50
Majority Classifier	33.98	31.83	32.91	38.15	32.25	35.20	31.15	25.28	28.22
Independent LSTMs	49.29	45.23	47.24	48.00	56.23	52.12	48.86	54.81	51.84
Independent SVMs	**57.25**	52.34	**54.80**	49.70	55.50	52.60	49.87	43.56	46.71
Window-Based SVMs	47.96	45.75	46.85	40.96	**59.26**	50.11	46.33	49.67	48.0
Cascading SVMs	55.87	53.01	54.44	51.34	56.17	53.76	50.42	45.48	47.95
9-Class SVM	51.82	53.21	52.52	49.62	53.06	51.34	47.94	48.09	48.02
Multi-Task LSTM	51.57	48.35	49.96	48.32	56.65	52.49	**53.63**	51.18	52.41
Seq2Seq	55.59	**53.86**	54.72	**54.46**	58.74	**56.60**	47.02	**59.21**	**53.12**

Table 5.6: Details for the performance of different models on each target in terms of F_{avg} (the columns) and the average over the target pairs. The highest score in each column is shown in bold.

5.5 Summary

To detect subjectivity expressed towards different targets, previous work often treats each target independently, ignoring the potential dependency that exists among the targets, the corresponding subjectivities, and other correlated hidden factors. For example, in an electoral social-media message, the stance expressed towards one candidate could be highly correlated with that towards another entity, and such a dependency exists widely in many other domains, including product reviews. In its difficult presence, subjectivity correlation could be associated with hidden factors such as topics under concern (e.g., two political candidates are not necessarily against each other on all topics), among others.

In this chapter, we relieve the independence assumption by jointly modeling the subjectivity expressed towards multiple targets. Particularly, we propose a framework that leverages recurrent neural models to capture the potentially complicated interaction between subjectivities expressed towards multiple targets. We experimentally show that the attention-based encoder-decoder framework is more effective in jointly modeling the overall position towards two related targets, compared to independent predictions of positions and other popular frameworks for joint learning, such as cascading classification and multi-task learning.

Additionally, We presented the first social media multi-target stance dataset of reasonable size for future exploration of the problem. In our dataset, each tweet is annotated for position towards more than one target. By making this dataset available, more work on joint learning of subjectivities corresponding to related targets is encouraged. This offers the opportunity of

bringing together sentiment analysis with textual inference and relation extraction.

Finally, we conducted several experiments to explore the advantages of modeling the interaction between stance labels in the output space by treating the task as a sequence-to-sequence learning. Furthermore, we tested other possible frameworks such as cascading classifiers, window-based classification and multi-task learning. The experiments demonstrated the effectiveness of the sequence-to-sequence neural networks in modeling a tweet together with the dependencies between subjectivities towards different targets of interest.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In this thesis, we addressed the task of automatic stance detection from social media texts. Stance detection was recently introduced as an important task in the area of opinion mining, with widespread applications in information retrieval and text summarization. Additionally, we investigated the problem of argument tagging to identify the reasons behinds one's position.

This thesis started with explaining our motivation for addressing this problem and the full description of tasks we explored. We explained the importance of social media opinion mining in various domains, including political elections, business intelligence, and medical decision making. We highlighted the differences between stance detection and sentiment analysis. Sentiment classification is formulated as determining whether a piece of text is positive, negative, or neutral. However, in stance detection, systems need to determine the position towards a given (pre-chosen) target of interest. The target of interest may not be explicitly mentioned in the text and it may not be the target of opinion in the text.

In prior works, stance classification was mostly applied on online debate forums data. These debates are two-sided and the data labels are often provided by the authors of the post. Here, we organized a first shared task competition on stance detection SemEval-2016, 'Task #6: Detecting Stance in Tweets' that received submissions from more than 19 teams in total. This fostered more research on stance detection and various models were developed for this task by deploying different techniques such as deep neural networks and distant supervision.

The rest of the thesis was organized in three main chapters presenting our proposed frameworks, methods and datasets.

In Chapter 3, we presented the first dataset of tweets annotated for both stance towards

given targets and sentiment. The tweets were also annotated for whether an opinion is expressed towards the given target or towards another entity. Subsequently, we used a linear-kernel SVM classifier that leveraged word and character n-grams as well as sentiment features drawn from available sentiment lexicons and word-embedding features drawn from additional unlabeled data. This simple, but effective, stance detection system obtained an F-score higher than the one obtained by the more complex, best-performing system in the SemEval competition. Finally, we conducted several experiments to tease out the interactions between the stance and sentiment. Notably, we showed that, even though sentiment features are useful for stance detection, they alone are not sufficient. We also showed that even though humans are capable of detecting stance towards a given target from texts that express opinion towards a different target, automatic systems perform poorly on such data.

In the next chapter, we investigated the problem of stance detection and the reasons behinds it in online news comments. We presented the first dataset of stance and arguments considering a particular medical study and following news that broadcasted it, as the target of interest. Furthermore, a new framework for argument tagging at document-level based on topic modeling (Non-Negative Matrix Factorization) was proposed. The main advantage of this framework is that it is minimally supervised and no labeled data is required for training, while, at the same time, it has comparable performance to multi-class, multi-output SVM classifier. Later, these automatically extracted argument tags were employed in the stance classifier as additional features. Experiments on Online News Comments showed the efficiency of these added features for both stance and its intensity classification.

Next, we explored the problem of multi-target stance detection in Twitter data. To detect stance expressed towards different targets, previous work often treats each target independently, ignoring the potential dependency that exists among the targets. In this research, we jointly modeled the subjectivity expressed towards multiple targets by leveraging neural models to capture the potentially complicated interaction between subjectivities expressed towards multiple targets. We experimentally showed that the attention-based encoder-decoder framework is more effective in jointly modeling the overall position towards two related targets, compared to independent predictions of positions and other models for joint learning, such as cascading classification and multi-task learning. Additionally, We presented the first social media multi-target stance dataset of reasonable size for future exploration of the problem.

To summarize, the key contributions of this research are as follows:

- We explored different features and algorithms for stance detection. We started from more traditional text classification approaches based on extracting different syntactic and semantic features from the text. Later, we investigated more advanced techniques such

as topic modeling and word embeddings. Finally, we applied state-of-the-art recurrent neural networks for stance classification, particularly we adopted an attentional encoder-decoder model for multi-target stance detection.

- We provided benchmark datasets for stance detection in social media. We created the first dataset of Twitter data labeled for both stance and sentiment. Then, we prepared the first dataset of online news comments labeled for stance and the reasons behind it. Lastly, we created the first dataset of tweets labeled for overall positions towards more than one target per post.

- We fostered more research on stance detection by conducting several experiments on this task, exploring the challenges and opportunities for future works. We showed that even though both stance and sentiment detection are framed as three-way classification tasks on a common dataset, automatic systems perform markedly better when detecting sentiment than when detecting stance towards a given target.

There are limitations to the proposed approaches. One popular way to express an opinion in social media is by using sarcasm and irony. In this research, we did not investigate the effect of using sarcasm and irony in opinion expression on stance detection. The other limitation of the proposed methods for stance detection is that they are all supervised approaches in which an annotated dataset of reasonable size for target/targets of interest is required. In the next section, we further explore the limitation of the proposed approaches and possible avenues to improve them.

6.2 Future Work

Several lines of research can be investigated in the future.

Transfer Learning: We are interested in developing stance detection systems that do not require stance-labeled instances for the target of interest, but instead, can learn from existing stance-labeled instances for other targets in the same domain. Hence, leveraging transfer-based learning is interesting to us, where a model trained on a target can be transferred to another related target. To this end, we can jointly model the content of the text and the targets of interest. This joint representation facilitates transferring models from one target to another one, by capturing the relationships between these targets. The other possibility to transfer a model trained for stance towards a particular target to another target of interest is to collect a small multi-target dataset such as the one we created in chapter 5, to bridge the gap between

these two targets by learning their relationship in different hidden topics. Another possibility to transfer a model trained for stance towards a particular target of interest to stance towards other related targets is to use generative adversarial networks (Goodfellow et al., 2014).

Inference: One of the key differences between sentiment and stance classification is that, in stance detection, the target of interest can be different from the target of opinion. As human beings, we can infer the position of an author towards the target of interest by considering the relationship between two entities given a context and a topic; but, as we showed in this research, automatic systems perform poorly on such data. We intent to focus more on these indirect or implicit opinions by modeling the ways in which stance is conveyed. Another possible avenues to improve the performance of the classifier on the data that express opinion towards a different target (not the target of interest) is to obtain more sophisticated features such as those derived from dependency parse trees (Sun et al., 2016) and automatically generated entity–entity relationship knowledge bases. Knowing that entity X is an adversary of entity Y can be useful in detecting stance towards Y from posts that express an opinion about X.

Tracking stance over time: Recently, it has been reported that companies can generate substantial revenue by tracking and analyzing their clients' positions towards them in social media. Tracking stance towards different candidates in political elections can also play a significant role in the success of an election campaign. Another direction of future work is to track how the distribution of stance towards a target changes over time.

Automatically extracting argument lists: Another avenue to pursue is to automatically extract the list of arguments. Currently, all similar research for argument tagging and reason identification have relied on a manually-extracted predefined list of tags for reason classification and argument tagging. However, manually creating such list for every domain is time-intensive and requires domain knowledge. Our proposed method to create the list of possible arguments is to first cluster posts based on the arguments they contain and to subsequently extract a representative sentence for each cluster (inspired from text summarization techniques). Another possibility for automatically creating the argument list is to follow similar approaches to automatic aspect extraction from product reviews.

Discourse structure: Discourse structure proved to be fundamentally important for sentiment analysis. We believe it is also effective for stance classification in social media, particularly for longer posts such as news comments. One of the challenges in automatically processing longer posts is noise and redundancy in data. For example, unlike tweets, in the commentsphere, there is no limitation in terms of the number of characters; therefore, online news comments are less focused compared to reviews and tweets, and the authors may write long posts that discuss different topics in a single post. Hence, parts of the posts might be

irrelevant in addressing the position towards the target of interest. Thus, information based on discourse structure can help by highlighting important text spans. Moreover, in the process of learning, those discourse relations that have more effect in the identification of stance can be recognized and assigned more weight.

Tree-Structure LSTM: Although promising results have been observed by applying chain-structured LSTM to many NLP tasks, other structures of LSTM, such as tree and directed acyclic graph, showed promising results for sentiment classification. Sentences in human languages are believed to be carried by not merely a linear sequence of words; instead, meaning is thought to interweave with structure. While a sequential application of LSTM may capture structural information implicitly, in practice, it sometimes lacks the claimed power. For example, even simply reversing the input sequences may result in significant differences in performance in tasks such as machine translation and speech recognition. One direction for future work is to employ a tree structure LSTM for stance classification, particularly for multi-target stance detection.

Alternative Classification Approaches: We plan to further improve the performance both on single and multi-target stance detection tasks by using alternative classification approaches such as ensemble learning. We also intent to adopt more advanced multi-task learning approaches for our task. For instance, the one suggested in Bilen and Vedaldi (2016) where different tasks interacts in a recurrent manner by updating the common shared representation by their predictions.

Bibliography

Rob Abbott, Marilyn Walker, Pranav Anand, Jean E Fox Tree, Robeson Bowmani, and Joseph King. How can you say such things?!?: Recognizing disagreement in informal political argument. In *Proceedings of the Workshop on Languages in Social Media*, pages 2–11. Association for Computational Linguistics, 2011.

Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery*, pages 36–43. ACM, 2005.

Amr Ahmed and Eric P Xing. Staying informed: supervised and semi-supervised multi-view topical analysis of ideological perspective. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1140–1150. Association for Computational Linguistics, 2010.

Camille Albert, Leila Amgoud, Florence Dupin de Saint-Cyr, Patrick Saint-Dizier, and Charlotte Costedoat. Introducing argumention in opinion analysis: Language and reasoning challenges. *Sentiment Analysis where AI meets Psychology (SAAIP)*, page 28, 2011.

Pranav Anand, Marilyn Walker, Rob Abbott, Jean E. Fox Tree, Robeson Bowmani, and Michael Minor. Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, pages 1–9, 2011.

Ion Androutsopoulos and Prodromos Malakasiotis. A survey of paraphrasing and textual entailment methods. *Journal of Artificial Intelligence Research*, pages 135–187, 2010.

Sanjeev Arora, Rong Ge, and Ankur Moitra. Learning topic models–going beyond svd. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pages 1–10. IEEE, 2012.

Isabelle Augenstein, Tim Rocktäschel, Andreas Vlachos, and Kalina Bontcheva. Stance detection with bidirectional conditional encoding. *arXiv preprint arXiv:1606.05464*, 2016a.

Isabelle Augenstein, Andreas Vlachos, and Kalina Bontcheva. USFD at SemEval-2016 Task 6: Any-Target Stance Detection on Twitter with Autoencoders. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016b.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

Luciano Barbosa and Junlan Feng. Robust sentiment detection on twitter from biased and noisy data. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 36–44. Association for Computational Linguistics, 2010.

Frank R Baumgartner, Suzanna L De Boef, and Amber E Boydstun. *The decline of the death penalty and the discovery of innocence*. Cambridge University Press, 2008.

Yoshua Bengio, Rejean Ducharme, and Pascal Vincent. A neural probabilistic language model. In *Advances in Neural Information Processing Systems*, 2001.

Luisa Bentivogli, Peter Clark, Ido Dagan, Hoa Dang, and Danilo Giampiccolo. The seventh PASCAL recognizing textual entailment challenge. In *Proceedings of Text Analysis Conference*, 2011.

Adam Bermingham and Alan Smeaton. On using twitter to monitor political sentiment and predict election results. In *Proceedings of the Workshop on Sentiment Analysis where AI meets Psychology*, pages 2–10, Chiang Mai, Thailand, 2011.

Philippe Besnard and Anthony Hunter. *Elements of argumentation*, volume 47. MIT press Cambridge, 2008.

Hakan Bilen and Andrea Vedaldi. Integrated perception with recurrent multi-task neural networks. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 235–243, 2016.

Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

Sasha Blair-Goldensohn, Kerry Hannan, Ryan McDonald, Tyler Neylon, George A Reis, and Jeff Reynar. Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era*, volume 14, 2008.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

Henrik Bøhler, Petter Asla, Erwin Marsi, and Rune Sætre. Idi@ntnu at semeval-2016 task 6: Detecting stance in tweets using shallow features and glove vectors for word representation. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 445–450, San Diego, California, June 2016.

Filip Boltuzic and Jan Šnajder. Back up your stance: Recognizing arguments in online discussions. In *Proceedings of the First Workshop on Argumentation Mining*, pages 49–58, 2014.

Samuel Brody and Noemie Elhadad. An unsupervised aspect-sentiment model for online reviews. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 804–812, 2010.

Tomáš Brychcın, Michal Konkol, and Josef Steinberger. Uwb: Machine learning approach to aspect-based sentiment analysis. In *Proceedings of the International Workshop on Semantic Evaluation*, Dublin, Ireland, August 2014.

Baptiste Chardon, Farah Benamara, Yannick Mathieu, Vladimir Popescu, and Nicholas Asher. Measuring the effect of discourse structure on sentiment analysis. In *Computational Linguistics and Intelligent Text Processing*, pages 25–37. Springer, 2013.

José M Chenlo, Alexander Hogenboom, and David E Losada. Rhetorical structure theory for polarity estimation: An experimental study. *Data & Knowledge Engineering*, 94:135–147, 2014.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014a.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014b.

François Chollet. Keras. `https://github.com/fchollet/keras`, 2015.

Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. End-to-end continuous speech recognition using attention-based recurrent nn: first results. *arXiv preprint arXiv:1412.1602*, 2014.

Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the International Conference on Machine Learning*, pages 160–167, 2008.

Michael Conover, Bruno Gonçalves, Jacob Ratkiewicz, Alessandro Flammini, and Filippo Menczer. Predicting the political alignment of twitter users. In *Proceedings of the IEEE International Conference on Privacy, Security, Risk, and Trust*, pages 192–199, 2011a.

Michael Conover, Jacob Ratkiewicz, Matthew R. Francisco, Bruno Gonçalves, Filippo Menczer, and Alessandro Flammini. Political polarization on twitter. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pages 89–96, 2011b.

Alexander Conrad, Janyce Wiebe, et al. Recognizing arguing subjectivity and argument tags. In *Proceedings of the Workshop on Extra-Propositional Aspects of Meaning in Computational Linguistics*, pages 80–88, 2012.

Mark Craven and Johan Kumlien. Constructing biological knowledge bases by extracting information from text sources. In *Proceedings of the Conference on Intelligent Systems for Molecular Biology*, pages 77–86, 1999.

Ido Dagan and Oren Glickman. Probabilistic textual entailment: Generic applied modeling of language variability. In *PASCAL workshop on Text Understanding and Mining*, pages 26–29, 2004.

Ido Dagan, Dan Roth, Mark Sammons, and Fabio Massimo Zanzotto. *Recognizing textual entailment: Models and applications*. Morgan & Claypool Publishers, 2013.

Kushal Dave, Steve Lawrence, and David M Pennock. Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web*, pages 519–528. ACM, 2003.

Scott Deerwester, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990.

Lingjia Deng and Janyce Wiebe. Sentiment propagation via implicature constraints. In *Proceedings of the Conference of the European Chapter of the Association for Computational Linguistics*, pages 377–385, Sweden, 2014.

Lingjia Deng and Janyce Wiebe. Joint prediction for entity/event-level sentiment analysis using probabilistic soft logic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, 2015a.

Lingjia Deng and Janyce Wiebe. Mpqa 3.0: An entity/event-level sentiment corpus. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Denver, Colorado, USA, 2015b.

Lingjia Deng, Janyce Wiebe, and Yoonjung Choi. Joint inference and disambiguation of implicit sentiments via implicature constraints. In *Proceedings of the International Conference on Computational Linguistics*, pages 79–88, 2014.

Marcelo Dias and Karin Becker. INF-UFRGS-OPINION-MINING at SemEval-2016 Task 6: Automatic Generation of a Training Corpus for Unsupervised Identification of Stance in Tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Xiaowen Ding, Bing Liu, and Philip S Yu. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 International Conference on Web Search and Data Mining*, pages 231–240. ACM, 2008.

Cícero Nogueira dos Santos and Bianca Zadrozny. Learning character-level representations for part-of-speech tagging. In *Proceedings of the 31th International Conference on Machine Learning ICML*, pages 1818–1826, June 2014.

Myroslava O. Dzikovska, Rodney D. Nielsen, and Claudia Leacock. The joint student response analysis and recognizing textual entailment challenge: making sense of student responses in educational applications. *Language Resources and Evaluation*, 50:67–93, 2016.

Douglas Eck and Jürgen Schmidhuber. Learning the long-term structure of the blues. In *Artificial Neural Networks–ICANN 2002*, pages 284–289. Springer, 2002.

Heba Elfardy and Mona Diab. CU-GWU at SemEval-2016 Task 6: Perspective at SemEval-2016 Task 6: Ideological Stance Detection in Informal Text. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

Manaal Faruqui, Jesse Dodge, Sujay K Jauhar, Chris Dyer, Eduard Hovy, and Noah A Smith. Retrofitting word vectors to semantic lexicons. *arXiv preprint arXiv:1411.4166*, 2014.

Atefeh Farzindar and Diana Inkpen. Natural language processing for social media. *Synthesis Lectures on Human Language Technologies*, 8(2):1–166, 2015.

Adam Faulkner. Automated classification of stance in student essays: An approach using stance target information and the Wikipedia link-based measure. In *Proceedings of the Flairs Conference*, 2014.

Ronen Feldman. Techniques and applications for sentiment analysis. *Communications of the ACM*, 56(4):82–89, 2013.

Austin Freeley and David Steinberg. *Argumentation and debate*. Cengage Learning, 2013.

Michael Gamon. Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In *Proceedings of the 20th international conference on Computational Linguistics*, page 841. Association for Computational Linguistics, 2004.

Debanjan Ghosh, Smaranda Muresan, Nina Wacholder, Mark Aakhus, and Matthew Mitsui. Analyzing argumentative discourse units in online interactions. In *Proceedings of the First Workshop on Argumentation Mining*, pages 39–48, 2014.

Kevin Gimpel, Nathan Schneider, Brendan O'Connor, Dipanjan Das, Daniel Mills, Jacob Eisenstein, Michael Heilman, Dani Yogatama, Jeffrey Flanigan, and Noah A. Smith. Part-of-speech tagging for twitter: Annotation, features, and experiments. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 42–47, Portland, Oregon, USA, June 2011.

Alec Go, Richa Bhayani, and Lei Huang. Twitter sentiment classification using distant supervision. Technical report, Stanford University, 2009.

Jennifer Golbeck and Derek Hansen. Computing political preference among twitter followers. In *Proceedings of the Conference on Human Factors in Computing Systems*, pages 1105–1108, New York, NY, 2011.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

Swapna Gottipati, Minghui Qiu, Yanchuan Sim, Jing Jiang, and Noah A. Smith. Learning topics and positions from Debatepedia. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1858–1868, Seattle, Washington, USA, October 2013.

Alex Graves. *Supervised sequence labelling with recurrent neural networks*. PhD thesis, Technische Universitat Munchen, 2008.

Alex Graves. *Supervised sequence labelling*. Springer, 2012.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Proceedings of the IEEE International Conference on Acoustics, speech and signal processing (icassp)*, pages 6645–6649. IEEE, 2013.

Stephan Greene and Philip Resnik. More than words: Syntactic packaging and implicit sentiment. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 503–511, 2009.

Ivan Habernal, Judith Eckle-Kohler, and Iryna Gurevych. Argumentation mining on the web from information seeking perspective. In *Proceedings of the Workshop on Frontiers and Connections between Argumentation Theory and Natural Language Processing*, pages 26–39, 2014.

Ben Hachey and Claire Grover. Automatic legal text summarisation: experiments with summary structuring. In *Proceedings of the 10th international conference on Artificial intelligence and law*, pages 75–84. ACM, 2005.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

Kazi Saidul Hasan and Vincent Ng. Extra-linguistic constraints on stance recognition in ideological debates. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 816–821, Sofia, Bulgaria, August 2013.

Kazi Saidul Hasan and Vincent Ng. Why are you taking this stance? identifying and classifying reasons in ideological debates. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 751–762, Doha, Qatar, October 2014.

Geremy Heitz, Stephen Gould, Ashutosh Saxena, and Daphne Koller. Cascaded classification models: Combining models for holistic scene understanding. In *Advances in Neural Information Processing Systems*, pages 641–648, 2009.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701, 2015.

Craig A Hill, Elizabeth Dean, and Joe Murphy. *Social Media, Sociality, and Survey Research*. John Wiley & Sons, 2013.

Jerry R Hobbs and Ellen Riloff. Information extraction. *Handbook of natural language processing*, 2, 2010.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, (8):1735–1780, 1997.

Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57. ACM, 1999.

Minqing Hu and Bing Liu. Mining and summarizing customer reviews. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 168–177, 2004.

Stephanie D. Husby and Denilson Barbosa. Topic classification of blog posts using distant supervision. In *Proceedings of the Workshop on Semantic Analysis in Social Media*, pages 28–36, 2012.

Ozan Irsoy and Claire Cardie. Deep recursive neural networks for compositionality in language. In Z. Ghahramani, M. Welling, C. Cortes, N.D. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2096–2104. Curran Associates, Inc., 2014a.

Ozan Irsoy and Claire Cardie. Opinion mining with deep recurrent neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 720–728, Doha, Qatar, October 2014b.

Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. Political ideology detection using recursive neural networks. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1113–1122, Baltimore, Maryland, June 2014.

Bruno Jakic. *Predicting sentiment of comments to news on Reddit*. PhD thesis, Intelligent Systems Lab Amsterdam, 2011.

Niklas Jakob and Iryna Gurevych. Extracting opinion targets in a single-and cross-domain setting with conditional random fields. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1035–1045. Association for Computational Linguistics, 2010.

Long Jiang, Mo Yu, Ming Zhou, Xiaohua Liu, and Tiejun Zhao. Target-dependent Twitter sentiment classification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics*, pages 151–160, 2011.

Mahesh Joshi and Carolyn Penstein-Rosé. Generalizing dependency features for opinion mining. In *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 313–316, Suntec, Singapore, August 2009.

Nobuhiro Kaji and Masaru Kitsuregawa. Building lexicon for sentiment analysis from massive collection of html documents. In *EMNLP-CoNLL*, pages 1075–1083, 2007.

Jaap Kamps, Maarten Marx, Robert J Mokken, and Maarten De Rijke. Using wordnet to measure semantic orientations of adjectives. In *LREC*, volume 4, pages 1115–1118, 2004.

Hiroshi Kanayama and Tetsuya Nasukawa. Fully automatic lexicon expansion for domain-oriented sentiment analysis. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 355–363. Association for Computational Linguistics, 2006.

Yoshikiyo Kato, Sadao Kurohashi, Kentaro Inui, Robert Malouf, and Tony Mullen. Taking sides: User classification for informal online political discourse. *Internet Research*, 18(2): 177–190, 2008.

Michael J Kearns and Umesh Virkumar Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

Alistair Kennedy and Diana Inkpen. Sentiment classification of movie reviews using contextual valence shifters. *Computational intelligence*, 22(2):110–125, 2006.

Svetlana Kiritchenko, Xiaodan Zhu, Colin Cherry, and Saif M. Mohammad. NRC-Canada-2014: Detecting aspects and sentiment in customer reviews. In *Proceedings of the International Workshop on Semantic Evaluation*, Dublin, Ireland, August 2014a.

Svetlana Kiritchenko, Xiaodan Zhu, and Saif M. Mohammad. Sentiment analysis of short informal texts. *Journal of Artificial Intelligence Research*, 50:723–762, 2014b.

Peter Krejzl and Josef Steinberger. Uwb at semeval-2016 task 6: Stance detection. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 408–412, San Diego, California, June 2016.

Lun-Wei Ku, Yu-Ting Liang, and Hsin-Hsi Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *AAAI spring symposium: Computational approaches to analyzing weblogs*, volume 100107, 2006.

Florian Kunneman, Christine Liebrecht, and Antal van den Bosch. The (un)predictability of emotional hashtags in Twitter. In *Proceedings of the Workshop on Language Analysis for Social Media*, pages 26–34, 2014.

Vasileios Lampos, Daniel Preotiuc-Pietro, and Trevor Cohn. A user-centric model of voting intention from social media. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 993–1003, 2013.

John Lawrence, Chris Reed, Colin Allen, Simon McAlister, Andrew Ravenscroft, and David Bourget. Mining arguments from 19th century philosophical texts using topic based modelling. *ACL 2014*, page 79, 2014.

Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, pages 556–562, 2001.

Joël Legrand and Ronan Collobert. Syntactic parsing of morphologically rich languages using deep neural networks. Technical report, Idiap, 2015.

Binyang Li, Lanjun Zhou, Shi Feng, and Kam-Fai Wong. A unified graph model for sentence-based opinion retrieval. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1367–1375. Association for Computational Linguistics, 2010.

Chenghua Lin and Yulan He. Joint sentiment/topic model for sentiment analysis. In *Proceedings of the 18th ACM conference on Information and knowledge management*, pages 375–384. ACM, 2009.

Chih-Jen Lin. Projected gradient methods for nonnegative matrix factorization. *Neural computation*, 19(10):2756–2779, 2007.

Wei-Hao Lin, Theresa Wilson, Janyce Wiebe, and Alexander Hauptmann. Which side are you on? Identifying perspectives at the document and sentence levels. In *Proceedings of the Conference on Computational Natural Language Learning*, pages 109–116, 2006.

Bing Liu. Sentiment analysis and subjectivity. *Handbook of natural language processing*, 2: 627–666, 2010.

Bing Liu. Sentiment analysis and opinion mining. *Synthesis Lectures on Human Language Technologies*, 5(1):1–167, 2012.

Bing Liu. *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press, 2015.

Bing Liu and Lei Zhang. A survey of opinion mining and sentiment analysis. In Charu C. Aggarwal and ChengXiang Zhai, editors, *Mining Text Data*, pages 415–463. Springer, 2012.

Can Liu, Wen Li, Bradford Demarest, Yue Chen, Sara Couture, Daniel Dakota, Nikita Haduong, Noah Kaufman, Andrew Lamont, Manan Pancholi, Kenneth Steimel, and Sandra Kübler. Iucl at semeval-2016 task 6: An ensemble model for stance detection in twitter. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 394–400, San Diego, California, June 2016.

Kang Liu, Liheng Xu, and Jun Zhao. Opinion target extraction using word-based translation model. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1346–1356. Association for Computational Linguistics, 2012.

Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-Yi Wang. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 912–921, Denver, Colorado, May–June 2015.

Marcus Liwicki, Alex Graves, Horst Bunke, and Jurgen Schmidhuber. A novel approach to online handwriting recognition based on bidirectional long short-term memory networks. In *In Proceedings of the 9th International Conference on Document Analysis and Recognition, ICDAR 2007*, 2007.

Yue Lu, Hongning Wang, ChengXiang Zhai, and Dan Roth. Unsupervised discovery of opposing opinion networks from forum discussions. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1642–1646. ACM, 2012.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*, 2015.

Walid Magdy, Hassan Sajjad, Tarek El-Ganainy, and Fabrizio Sebastiani. Bridging social media via distant supervision. *Social Network Analysis and Mining*, 5(1):1–12, 2015.

Marco Marelli, Luisa Bentivogli, Marco Baroni, Raffaella Bernardi, Stefano Menini, and Roberto Zamparelli. SemEval-2014 Task 1: Evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In *Proceedings of the International Workshop on Semantic Evaluation*, 2014.

Justin Martineau and Tim Finin. Delta tfidf: An improved feature space for sentiment analysis. In *ICWSM*, 2009.

Diana Maynard and Adam Funk. Automatic detection of political opinions in tweets. In *Proceedings of the ESWC Workshop on the Semantic Web*, pages 88–99, 2011.

Qiaozhu Mei, Xu Ling, Matthew Wondra, Hang Su, and ChengXiang Zhai. Topic sentiment mixture: modeling facets and opinions in weblogs. In *Proceedings of the 16th international conference on World Wide Web*, pages 171–180. ACM, 2007.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013a.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119, 2013b.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. Linguistic regularities in continuous space word representations. In *HLT-NAACL*, pages 746–751, 2013c.

Anthony B Miller, Claus Wall, Cornelia J Baines, Ping Sun, Teresa To, Steven A Narod, et al. Twenty five year follow-up for breast cancer incidence and mortality of the canadian national breast screening study: randomised screening trial. *Bmj*, 348, 2014.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, 2009.

Amita Misra, Brian Ecker, Theodore Handleman, Nicolas Hahn, and Marilyn Walker. nldsucsc at SemEval-2016 Task 6: A Semi-Supervised Approach to Detecting Stance in Tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Andriy Mnih and Geoffrey Hinton. Three new graphical models for statistical language modelling. In *Proceedings of the 24th international conference on Machine learning*, pages 641–648. ACM, 2007.

Andriy Mnih and Geoffrey E. Hinton. A scalable hierarchical distributed language model. In *Advances in Neural Information Processing Systems*, pages 1081–1088, 2009.

Volodymyr Mnih, Nicolas Heess, Alex Graves, et al. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems*, pages 2204–2212, 2014.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 31–41, San Diego, California, June 2016a.

Saif Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. A dataset for detecting stance in tweets. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 3945–3952, may 2016b.

Saif M Mohammad. Sentiment analysis: Detecting valence, emotions, and other affectual states from text. *Emotion Measurement*, pages 201–238.

Saif M. Mohammad. #emotional tweets. In *Proceedings of the Joint Conference on Lexical and Computational Semantics*, pages 246–255, Montréal, Canada, 2012.

Saif M. Mohammad. A practical guide to sentiment annotation: Challenges and solutions. In *Proceedings of the Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 2016.

Saif M. Mohammad and Peter D. Turney. Emotions evoked by common words and phrases: Using Mechanical Turk to create an emotion lexicon. In *Proceedings of the Workshop on Computational Approaches to Analysis and Generation of Emotion in Text*, pages 26–34, 2010.

Saif M. Mohammad, Bonnie J. Dorr, Graeme Hirst, and Peter D. Turney. Computing lexical contrast. *Computational Linguistics*, 39(3):555–590, 2013a.

Saif M. Mohammad, Svetlana Kiritchenko, and Xiaodan Zhu. NRC-Canada: Building the state-of-the-art in sentiment analysis of tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, Atlanta, Georgia, USA, June 2013b.

Saif M. Mohammad, Xiaodan Zhu, Svetlana Kiritchenko, and Joel Martin. Sentiment, emotion, purpose, and style in electoral tweets. *Information Processing and Management*, 51: 480–499, 2015.

Saif M. Mohammad, Svetlana Kiritchenko, Parinaz Sobhani, Xiaodan Zhu, and Colin Cherry. SemEval-2016 Task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation*, San Diego, California, 2016c.

Saif M. Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. *Special Section of the ACM Transactions on Internet Technology on Argumentation in Social Media*, In Press, 2016d.

Rodrigo Moraes, JoãO Francisco Valiati, and Wilson P GaviãO Neto. Document-level sentiment classification: An empirical comparison between svm and ann. *Expert Systems with Applications*, 40(2):621–633, 2013.

Alejandro Moreo, M Romero, JL Castro, and Jose Manuel Zurita. Lexicon-based comments-oriented news sentiment analyzer system. *Expert Systems with Applications*, 39(10):9166–9180, 2012.

Arjun Mukherjee and Bing Liu. Mining contentions from discussions and debates. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 841–849, 2012.

Arjun Mukherjee and Bing Liu. Discovering user interactions in ideological discussions. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 671–681, Sofia, Bulgaria, August 2013.

Tony Mullen and Nigel Collier. Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 412–418, Barcelona, Spain, July 2004.

Tony Mullen and Robert Malouf. A preliminary investigation into sentiment analysis of informal political discourse. In *AAAI Spring Symposium: Computational Approaches to Analyzing Weblogs*, pages 159–162, 2006.

Akiko Murakami and Rudy Raymond. Support or oppose?: classifying positions in online debates from reply activities and opinion expressions. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 869–875. Association for Computational Linguistics, 2010.

Jin-cheon Na, Haiyang Sui, Christopher Khoo, Syin Chan, and Yunyun Zhou. Effectiveness of simple linguistic processing in automatic sentiment classification of product reviews. In *In Proceeding of the Conference of the International Society for Knowledge Organization*, 2004.

David Newman, Chaitanya Chemudugunta, Padhraic Smyth, and Mark Steyvers. Analyzing entities and topics in news articles using statistical topic models. In *Intelligence and Security Informatics*, pages 93–104. Springer, 2006.

Vincent Ng, Sajib Dasgupta, and SM Arifin. Examining the role of linguistic knowledge sources in the automatic identification and classification of reviews. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 611–618. Association for Computational Linguistics, 2006.

Viet-An Nguyen, Jordan L Boyd-Graber, and Philip Resnik. Lexical and hierarchical topic regression. In *Advances in Neural Information Processing Systems*, pages 1106–1114, 2013.

Georgios Paltoglou and Mike Thelwall. A study of information retrieval weighting schemes for sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1386–1395. Association for Computational Linguistics, 2010.

Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 115–124. Association for Computational Linguistics, 2005.

Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135, 2008.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics, 2002.

Ankur P. Parikh, Hoifung Poon, and Kristina Toutanova. Grounded semantic parsing for complex knowledge extraction. In *Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2015.

Braja Gopal Patra, Dipankar Das, and Sivaji Bandyopadhyay. JU_NLP at SemEval-2016 Task 6: Detecting Stance in Tweets using Support Vector Machines. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Fabian Pedregosa, Gaël Varoquaux, et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. *Proceedings of the Empiricial Methods in Natural Language Processing (EMNLP 2014)*, 12:1532–1543, 2014.

Richard M Perloff. *The dynamics of persuasion: communication and attitudes in the twenty-first century*. Routledge, 2010.

Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. SemEval-2014 Task 4: Aspect based sentiment analysis. In *Proceedings of the International Workshop on Semantic Evaluation*, Dublin, Ireland, August 2014.

Maria Pontiki, Dimitrios Galanis, Haris Papageogiou, Suresh Manandhar, and Ion Androutsopoulos. SemEval-2015 Task 12: Aspect based sentiment analysis. In *Proceedings of the International Workshop on Semantic Evaluation*, Denver, Colorado, 2015.

Martin Potthast, Benno Stein, Fabian Loose, and Steffen Becker. Information retrieval in the commentsphere. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 3(4):68, 2012.

Ashwin Rajadesingan and Huan Liu. Identifying users with opposing opinions in Twitter debates. In *Proceedings of the Conference on Social Computing, Behavioral-Cultural Modeling and Prediction*, pages 153–160. Washington, DC, USA, 2014.

Marta Recasens, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. Linguistic models for analyzing and detecting biased language. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1650–1659, 2013.

Kevin Reschke, Martin Jankowiak, Mihai Surdeanu, Christopher D. Manning, and Daniel Jurafsky. Event extraction using distant supervision. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 4527–4531, 2014.

Philip Resnik, William Armstrong, Leonardo Claudino, Thang Nguyen, Viet-An Nguyen, and Jordan Boyd-Graber. Beyond lda: Exploring supervised topic modeling for depression-related language in twitter. In *Proceedings of the 2nd Workshop on Computational Linguistics and Clinical Psychology (CLPsych)*, 2015.

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. Named entity recognition in tweets: an experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, 2011.

Sara Rosenthal, Preslav Nakov, Svetlana Kiritchenko, Saif M. Mohammad, Alan Ritter, and Veselin Stoyanov. SemEval-2015 Task 10: Sentiment analysis in Twitter. In *Proceedings of the International Workshop on Semantic Evaluations*, 2015.

Jodi Schneider, Brian Davis, and Adam Wyner. Dimensions of argumentation in social media. In *Knowledge Engineering and Knowledge Management*, pages 21–25. Springer, 2012.

Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

Farial Shahnaz, Michael W Berry, V Paul Pauca, and Robert J Plemmons. Document clustering using nonnegative matrix factorization. *Information Processing & Management*, 42(2):373–386, 2006.

Parinaz Sobhani, Herna Viktor, and Stan Matwin. Learning from imbalanced data using ensemble methods and cluster-based undersampling. In *New Frontiers in Mining Complex Patterns - Third International Workshop, Held in Conjunction with ECML-PKDD, Revised Selected Papers*, pages 69–83, September 2014.

Parinaz Sobhani, Diana Inkpen, and Stan Matwin. From argumentation mining to stance classification. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 67–77, Denver, CO, June 2015.

Parinaz Sobhani, Saif Mohammad, and Svetlana Kiritchenko. Detecting stance in tweets and analyzing its interaction with sentiment. In *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pages 159–169, Berlin, Germany, August 2016.

Parinaz Sobhani, Xiaodan Zhu, and Diana Inkpen. A dataset for multi-target stance detection. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, Valencia, Spain, April 2017.

Richard Socher, Eric H Huang, Jeffrey Pennin, Christopher D Manning, and Andrew Y Ng. Dynamic pooling and unfolding recursive autoencoders for paraphrase detection. In *Advances in Neural Information Processing Systems*, pages 801–809, 2011a.

Richard Socher, Jeffrey Pennington, Eric Huang, Andrew Y. Ng, and Christopher D. Manning. Semi-supervised recursive autoencoders for predicting sentiment distributions. In *Conference on Empirical Methods in Natural Language Processing*, 2011b.

Richard Socher, Alex Perelygin, Jean Y. Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '13, Seattle, USA, 2013. Association for Computational Linguistics.

Swapna Somasundaran and Janyce Wiebe. Recognizing stances in online debates. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 226–234, Suntec, Singapore, 2009a.

Swapna Somasundaran and Janyce Wiebe. Recognizing stances in online debates. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 226–234, 2009b.

Swapna Somasundaran and Janyce Wiebe. Recognizing stances in ideological on-line debates. In *Proceedings of the NAACL HLT 2010 Workshop CAAGET*, pages 116–124, 2010.

Swapna Somasundaran, Josef Ruppenhofer, and Janyce Wiebe. Detecting arguing and sentiment in meetings. In *Proceedings of the SIGdial Workshop on Discourse and Dialogue*, volume 6, 2007.

Radu Soricut and Franz Och. Unsupervised morphology induction using word embeddings. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1627–1637, Denver, Colorado, May–June 2015.

Dhanya Sridhar, Lise Getoor, and Marilyn Walker. Collective stance classification of posts in online debate forums. In *Proceedings of the Joint Workshop on Social Dynamics and Personal Attributes in Social Media*, pages 109–117, Baltimore, Maryland, June 2014.

Dhanya Sridhar, James Foulds, Bert Huang, Lise Getoor, and Marilyn Walker. Joint models of disagreement and stance in online debate. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 116–125, Beijing, China, July 2015.

Philip J Stone, Robert F Bales, J Zvi Namenwirth, and Daniel M Ogilvie. The general inquirer: A computer system for content analysis and retrieval based on the sentence as a unit of information. *Behavioral Science*, 7(4):484–498, 1962.

Qingying Sun, Zhongqing Wang, Qiaoming Zhu, and Guodong Zhou. Exploring various linguistic features for stance detection. In *International Conference on Computer Processing of Oriental Languages*, pages 840–847. Springer, 2016.

Suhuan Sun, Gongsheng Kong, and Changwei Zhao. Polarity words distance-weight count for opinion analysis of online news comments. *Procedia Engineering*, 15:1916–1920, 2011.

Ilya Sutskever, James Martens, and Geoffrey E Hinton. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pages 1017–1024, 2011.

Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

Reid Swanson, Brian Ecker, and Marilyn Walker. Argument mining: Extracting arguments from online dialogue. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 217, 2015.

Maite Taboada, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon-based methods for sentiment analysis. *Computational linguistics*, 37(2):267–307, 2011.

Duyu Tang, Furu Wei, Nan Yang, Ming Zhou, Ting Liu, and Bing Qin. Learning sentiment-specific word embedding for Twitter sentiment classification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1555–1565, 2014.

Simone Teufel, Advaith Siddharthan, and Colin Batchelor. Towards discipline-independent argumentative zoning: evidence from chemistry and computational linguistics. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3*, pages 1493–1502. Association for Computational Linguistics, 2009.

Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.

Matt Thomas, Bo Pang, and Lillian Lee. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 327–335, 2006.

Ivan Titov and Ryan McDonald. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of ACL-08: HLT*, pages 308–316, Columbus, Ohio, June 2008a.

Ivan Titov and Ryan McDonald. Modeling online reviews with multi-grain topic models. In *Proceedings of the 17th international conference on World Wide Web*, pages 111–120. ACM, 2008b.

Stephen E Toulmin. *The uses of argument*. Cambridge University Press, 2003.

Andranik Tumasjan, Timm O. Sprenger, Philipp G. Sandner, and Isabell M. Welpe. Election forecasts with Twitter: How 140 characters reflect the political landscape. *Social Science Computer Review*, 29(4):402–418, 2010.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: a simple and general method for semi-supervised learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 384–394, 2010.

Peter Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424, Philadelphia, Pennsylvania, USA, July 2002.

Peter D Turney and Michael L Littman. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems (TOIS)*, 21(4):315–346, 2003.

Martin Tutek, Ivan Sekulić, Paula Gombar, Ivan Paljak, Filip Čulinović, Filip Boltužić, Mladen Karan, Domagoj Alagić, and Jan Šnajder. TakeLab at SemEval-2016 Task 6: Stance Classification in Tweets Using a Genetic Algorithm Based Ensemble. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Prashanth Vijayaraghavan, Ivan Sysoev, Soroush Vosoughi, and Deb Roy. Deepstance at semeval-2016 task 6: Detecting stance in tweets using character and word-level cnns. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 413–419, San Diego, California, June 2016.

Oriol Vinyals, Łukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. In *Advances in Neural Information Processing Systems*, pages 2773–2781, 2015a.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2015b.

Marilyn A Walker, Pranav Anand, Rob Abbott, Jean E Fox Tree, Craig Martell, and Joseph King. That is your evidence?: Classifying stance in online political debate. *Decision Support Systems*, 53(4):719–729, 2012a.

Marilyn A. Walker, Pranav Anand, Robert Abbott, and Ricky Grant. Stance classification using dialogic properties of persuasion. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 592–596, 2012b.

Marilyn A. Walker, Jean E. Fox Tree, Pranav Anand, Rob Abbott, and Joseph King. A corpus for research on deliberation and debate. In *Proceedings of the International Conference on Language Resources and Evaluation*, pages 812–817, 2012c.

Hanna M Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984. ACM, 2006.

Lu Wang and Claire Cardie. Improving agreement and disagreement identification in online discussions with a socially-tuned sentiment lexicon. In *Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 97–106, Baltimore, Maryland, June 2014.

Wan Wei, Xiao Zhang, Xuqin Liu, Wei Chen, and Tengjiao Wang. pkudblab at SemEval-2016 Task 6: A Specific Convolutional Neural Network System for Effective Stance Detection. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Janyce M Wiebe, Rebecca F Bruce, and Thomas P O'Hara. Development and use of a gold-standard data set for subjectivity classifications. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 246–253. Association for Computational Linguistics, 1999.

Theresa Wilson and Janyce Wiebe. Annotating attributions and private states. In *Proceedings of the Workshop on Frontiers in Corpus Annotations II: Pie in the Sky*, pages 53–60. Association for Computational Linguistics, 2005.

Theresa Wilson, Paul Hoffmann, Swapna Somasundaran, Jason Kessler, Janyce Wiebe, Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. Opinionfinder: A system for subjectivity analysis. In *Proceedings of hlt/emnlp on interactive demonstrations*, pages 34–35, 2005a.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 347–354, Vancouver, British Columbia, Canada, 2005b.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 347–354. Association for Computational Linguistics, 2005c.

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity: An exploration of features for phrase-level sentiment analysis. *Computational linguistics*, 35 (3):399–433, 2009.

Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Sara Rosenthal, Veselin Stoyanov, and Alan Ritter. SemEval-2013 Task 2: Sentiment analysis in Twitter. In *Proceedings of the International Workshop on Semantic Evaluation*, Atlanta, USA, June 2013a.

Theresa Wilson, Zornitsa Kozareva, Preslav Nakov, Sara Rosenthal, Veselin Stoyanov, and Alan Ritter. SemEval-2013 Task 2: Sentiment analysis in Twitter. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '13, Atlanta, Georgia, USA, June 2013b.

Rui Xia and Chengqing Zong. Exploring the use of word relation features for sentiment classification. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1336–1344, 2010.

Rui Xia and Chengqing Zong. A pos-based ensemble model for cross-domain sentiment classification. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 614–622, Chiang Mai, Thailand, November 2011.

Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273. ACM, 2003.

Tae Yano, Philip Resnik, and Noah A Smith. Shedding (a thousand points of) light on biased language. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 152–158, 2010.

Ainur Yessenalina and Claire Cardie. Compositional matrix-space models for sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 172–182, 2011.

Igarashi Yuki, Komatsu Hiroya, Kobayashi Sosuke, Okazaki Naoaki, and Inui Kentaro. Tohoku at SemEval-2016 Task 6: Feature-based Model versus Convolutional Neural Network for Stance Detection. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Guido Zarrella and Amy Marsh. MITRE at SemEval-2016 Task 6: Transfer Learning for Stance Detection. In *Proceedings of the International Workshop on Semantic Evaluation*, SemEval '16, San Diego, California, June 2016.

Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

Ying Zhang, Yi Fang, Xiaojun Quan, Lin Dai, Luo Si, and Xiaojie Yuan. Emotion tagging for comments of online news by meta classification with heterogeneous information sources. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, pages 1059–1060. ACM, 2012.

Zhihua Zhang and Man Lan. Ecnu at semeval 2016 task 6: Relevant or not? supportive or not? a two-step learning system for automatic detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 451–457, San Diego, California, June 2016.

Changwei Zhao, Heng Liu, Qinke Peng, and Ying Yang. Free-tagging methods for opinion analysis of online news comments. In *Control and Automation (ICCA), 2010 8th IEEE International Conference on*, pages 1190–1195. IEEE, 2010a.

Wayne Xin Zhao, Jing Jiang, Hongfei Yan, and Xiaoming Li. Jointly modeling aspects and opinions with a maxent-lda hybrid. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 56–65. Association for Computational Linguistics, 2010b.

Jie Zhou and Wei Xu. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 1127–1137, Beijing, China, July 2015.

Jie Zhou, Chen Lin, and Bi-cheng Li. Research of sentiment classification for net news comments by machine learning. *Journal of Computer Applications*, 30(4):1011–1014, 2010.

Xiaodan Zhu, Hongyu Guo, and Parinaz Sobhani. Neural networks for integrating compositional and non-compositional sentiment in sentiment composition. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 1–9, Denver, Colorado, June 2015a.

Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. Long short-term memory over recursive structures. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 1604–1612, July 2015b.

Xiaodan Zhu, Parinaz Sobhani, and Hongyu Guo. Dag-structured long short-term memory for semantic compositionality. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 917–926, San Diego, California, June 2016.