UNIVERSITY OF CALIFORNIA, SAN DIEGO

Identifying Author Topic Stance in Online Discussion Forums

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Linguistics

by

Gary Patterson

Committee in charge:

Professor Andrew Kehler, Chair
Professor Ivano Caponigro
Professor Robert Malouf
Professor Lawrence Saul
Professor Eva Wittenberg

2018

The dissertation of Gary Patterson is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____
Chair

University of California, San Diego

2018

DEDICATION

To Greg and Wiki

EPIGRAPH

Marriage equality is a hustler's feeding frenzy of gold-diggers. I campaigned for marriage equality in Maryland because I believe we should have the right to it, but I personally don't want to get married. I don't want to imitate the traditions of heterosexual people. I hate weddings; they make me uneasy.

— Anonymous Internet Commenter

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to begin by thanking the members on my committee for their thoughtful discussions and guidance during the duration of this project. First and foremost, I thank Andy for his solid, pragmatic advice all the way through my graduate school experience, and for his infinite patience with me, especially when my progress seemed incremental or stalled. I am very grateful to Rob and Lawrence for their wise technical expertise, modeling suggestions, and warnings of pitfalls to watch out for. Many thanks to Eva for her interest and enthusiasm in the project, and some motivating chats along the way. To Ivano, I cannot thank you enough for your mentorship and friendship, and the perspective that you helped me maintain during this dragon-slaying endeavor. I also want to thank Roger Levy for his time on the committee while he was still in San Diego, and for his valuable input during the formative stages of this project. My deep and abiding gratitude goes to my research assistant on this project, Laura Dentz, who willingly submerged herself into the murky underworld of internet comments. Also, I give major thanks to my other tireless annotators: Jasmeen Kanwal, Gwen Gillingham, Barath Ezhilan, and Garth Kimbrell. It was back-breaking work to be sure, but also kind of fun in a twisted way, I hope.

The UCSD Linguistics department has been a nurturing and stimulating home for the past few years, full of smart, dedicated, and intellectually generous folks. Particular thanks to Eric Bakovic, the friendly, neighborhood DGS, and to Sharon Rose for her unflappable chair-iness, capably steering the department along its course. I'd also love to acknowledge all of the underappreciated, behind-the-scenes tasks that Alycia Randol, Rachel Pekras, and Lucie Wiseman do to keep the department machinery running so smoothly, and for their help in my navigating the university's many administrative idiosyncracies.

I can't let this opportunity pass without giving a shout out to my class cohort: Younah Chung, Gustavo Guajardo, Mark Myslin, Page Piccinini, and Nadav Sofer. It seems as if Larry the Larynx and all of those tantalizing syntax homework assignments happened a lifetime ago, but I'm happy that we went through all that stuff together and came out the other side better and stronger for the experience. Other graduate students and friends have helped me no end with long discussions, technical assistance, moments

of insight, and more. To Vinod Prabhakaran, Rebekah Baglini, Emily Morgan, Bethany Keffala, Rodolfo Mata, Gabriel Doyle, Kati Hout, Savi Namboodiripad, Till Poppels, Jamie Alexandre, and about thousand others: I sincerely thank you all for enriching my work and my experience at UCSD.

Most of all, I want to acknowledge the love and support of my family, especially my parents, Joyce and Cavan, for believing in me and encouraging my academic pursuits from the very beginning. And finally to Greg, for putting up with all this craziness for all this time: I really couldn't have done this without your support.

VITA

1993	B.S. in Mathematics with Statistics *First Class Honors*, University of Bristol
2008	M.A. in Linguistics, San Francisco State University, San Francisco
2018	Ph.D. in Linguistics, University of California, San Diego

PUBLICATIONS

Patterson, G. and Kehler, A., "Predicting the Presence of Discourse Connectives", *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013.

Patterson, G. and Caponigro, I., "The Puzzling Degraded Status of *who* Free Relatives in English", *English Language and Linguistics*, 20(2), 2016.

ABSTRACT OF THE DISSERTATION

Identifying Author Topic Stance in Online Discussion Forums

by

Gary Patterson

Doctor of Philosophy in Linguistics

University of California, San Diego, 2018

Professor Andrew Kehler, Chair

A standard feature of the contemporary internet landscape is the ability for people to comment on published content and to interact with other individuals, discussing the issues at hand and engaging with each other in debate. In this thesis, I describe a method for the automatic detection of author stances in online forums with respect to discussions on divisive, polarizing social issues, such as gun control and marriage equality – a task which is often unproblematic for human readers of the discourse. The research investigates the linguistic and rhetorical devices used by discussion participants to express their topic stance in the context of multi-party, multi-threaded discourse. Along the way, I address necessary sub-tasks in the author stance detection problem, such as the classification of the topic stance of an individual contribution to the discourse, and the assessment of the level of agreement or disagreement between adjacent posts - which is crucial, given the

highly interactive nature of this genre. I also identify features that provide evidence of an author's topic stance from the very structure of the discourse, without any information at all from the text of the comments posted. The final model is a collective classifier that is able to synthesize all of the stance indicators provided by these different sources, deal with the inconsistencies in this information that may arise, and arrive at a single prediction of the topic stance for every participant in the discussion. The model has many applications in industry and public life, including more tailored newsfeeds, social network suggestions, and use in political fundraising or advocacy campaigns.

Part I

Introduction and Background

Chapter 1

Introduction

1.1 Overview

Given the rapid and enormous advances in internet technology over the past decade, our culture has witnessed an explosion in the amount of user-generated content on the web, especially in the context of social media, where people create online communities to share information, ideas, opinions, and personal messages. One rich source of such content relates to the discussion of news stories and current events on news media websites or other online discussion forums. For a given story or news article, large numbers of users can participate in an extended discussion, in which they express their opinions on the topic at hand, and actively engage with other participants in the discourse.

The abundance of this kind of naturally-occurring, user-generated dialogic content gives researchers a new tool to study how interlocutors interact, and to analyze how they express their opinions on topics under discussion. In particular, the data from such online debates allows us to consider the process of *stance-taking* - namely, the linguistic-, discourse-, and rhetorical strategies that discourse participants use to position themselves (*in favor of* or *against*) towards a topic or proposition under discussion. Here, the target of the stance can be a political issue (such as whether the UK should withdraw from the European Union), an ideological one (e.g. whether euthanasia ought to be legalized), an individual or entity (e.g what is one's opinion of Donald Trump), or a simple proposition (whether, say, Batman is better than Superman). The task of automatically determining from a text whether the author of the text is in favor of or against a given target of

interest is called *stance detection*.

This thesis describes a method for the automatic detection of author stance in online discussion forums. I focus specifically the comment sections of news websites, and the task is constrained to consider only polarizing ideological topics around politics or divisive social issues (abortion, gun control, same sex marriage, and so on). Such cases are generally characterizable as *two-sided* debates, in which the position of a discourse participant can more easily be categorized as falling on one side of the issue or the other. In the course of presenting an automatic stance detection system, I identify and discuss the most reliable features that allow for this classification.

The ability to classify the stance of users in online communities with respect to various topics - and thereby to identify groups of users with similar opinions - has many applications in industry, including better recommender systems, newsfeeds that are more tailored to individual user preferences, friend or contact suggestions in social networks, and more targeted marketing and promotional activities. Further, if particular topic stances are correlated with, and can be mapped to, social or political ideologies, the automatic detection of topic stance also has applications for non-profit and governmental agencies, such as being used to target get-out-the-vote efforts, direct political fundraising, or advocacy campaigns.

In the remainder of this introductory chapter, I first provide an overview of the domain of online discussions. I then describe the typical features of discourse in this genre, and walk through an example that exemplifies these characteristics, pointing out why this makes automatic stance detection a challenging task. I also briefly describe the task of stance detection and contrast it to the related topic of sentiment analysis. The chapter concludes with a roadmap for the rest of the thesis.

1.2 Online discussion forums

Online comments have become an essential component of a successful news publication, with over 90% of news organizations allowing reader comments on all or some of their online articles (Goodman and Cherubini (2013)). Readers feel that they have the right to express their opinions on the topics of the day and interact with other readers in

a dialog-based online discussion forum, and when readers are not able to comment on articles or have this ability taken away, they are not happy. This was recently evidenced by the uproar (played out in other online forums, naturally) raised by readers of the science and technology magazine *Popular Science* when it announced in September 2013 that it would no longer be accepting comments on new articles on its website. The news organizations themselves also benefit from online commenting: comments by informed readers can add insight and additional perspective to the articles, foster a sense of community among discussion participants and result in a more loyal readership who visit their site more frequently and stay longer during each visit. While it is true that recently a number of news websites have limited or even removed their commenting platforms (on the grounds that it is too costly to moderate these boards and ensure that offensive or hateful speech is removed), it does appear that online commenting will be an integral part of the cultural landscape going forward.

The multi-party, multi-threaded nature of discussions among online forum participants results in a richly structured dialog structure that is different from more traditional face-to-face, multi-party conversations. The technological afforandances of commenting platforms typically allow for a discourse participant either to initiate a new thread of conversation (i.e. by posting a 'root-level' comment[1] addressing a point raised in the main article) or to post a response to a comment in an existing thread (addressing a point raised in the previous comment, or expressing an opinion directed at the author of that comment). With respect to the typical mix of 'root level' and 'reply' comments in a discussion, I found that in a sample of 100,000 posts scraped from comment threads on www.cnn.com, some 74% were posts responding to other comments, and hence only 26% were written directly in response to the article itself. Relatedly, there can be a wide range in the number of reply posts that a comment garners. There will often be a small number of thoughtful, well-written (or, on the other hand, incendiary) posts earlier on in the discussion that generate a large number of comments in direct response. However,

[1]In this dissertation I will use the terms 'post' and 'comment' interchangeably to describe a contribution by a participant (or 'commenter') to an online discussion forum. As a result, a tree-like discourse structure is generated, in which a comment by one author can generate one or more responses from other conversation participants, which in turn each can provoke responses of their own.

Posts can range in length from a single word, orthographic token, or emoticon (e.g. 'Right', '?', ':)') to a multi-paragraph passage.

many comments receive just a single reply (indicative of two participants involved in a back-and-forth conversation amidst the multi-party discourse) and it is not atypical for as many as 50% of comments to not generate any responses at all. In all, for a given discussion under a particular article on the news website, there can be many active conversation threads unfolding over time as the discussion progresses, each cascading from a different 'root' comment, and then branching off with every new response that a comment gets. Active users may find themselves interacting with the same commenters across different conversation threads simultaneously.

Another characteristic aspect of the discourse structure of internet comments is the distribution of the number of posts made by the discussion participants. Rather than everybody contributing equally to the discussion, we typically observe a long tail distribution, where there is a small number of 'super users' dominating the conversation, contributing the majority of the comments on any particular article. On other other hand, most participants take a much less active role in the discussion, contributing just a comment or two. In many conversations, this long tail can represent as much as one half of all the discourse participants. This distribution poses certain challenges for the author stance classification task, in that for many participants we will only have a very limited about of data from which to infer their position.

The resulting discouse is an interesting hybrid, with properties of both written and spoken modalities. On one hand, internet comments are often short, generally informal and full of subjective or emotional language, and are frequently fragmented phrases rather than complete grammatical sentences. In this way, they are characteristic of spontaneous spoken communication. Also, if a discussion on a new article unfolds in real-time this can result in instantaneous dyadic flurries between pairs of commenters, much like an informal face-to-face conversation. On the other hand, since the contributions to the discourse are indeed written and permanent, discussions can also be asynchronous, unfolding over the course of a few days or longer. Commenters are able to take time to compose and revise their contribution before posting, resulting in thoughtful, edited position statements on a topic intended for a broader audience to be read at a later date. Moreover, discussion participants can easily copy and paste excerpts of posts left by previous commenters and provide commentary on them, resulting in a higher rate of 'meta' comments (i.e.

comments about comments).

1.3 A prototypical example of online discourse

The typical characteristics of a discourse that occurs between multiple participants in an online forum are best illustrated with a concrete example. A close reading of one such example will also serve to show the complexity of the automatic author stance detection task, as well as give some hints on how this problem may be tackled. Figure 1.1 shows a discourse excerpt from an online discussion on gun control taking place on a news website in response to an article on that topic.[2]

This online conversation comprises a total of eight comments made by four discourse participants (three comments each written by two authors, and a single comment each from two other contributors). It begins with comment C_1, written by the author *ConservativeKen*, in direct response to a stimulus news article - in this case, regarding some proposals for stricter requirements on gun ownership that the democratic caucus in the U.S. House of Representatives was formulating. The author of comment C_1 expresses his (negative) stance towards the issue of tighter gun control legislation clearly and unambiguously, using topic-specific language (*'It is my [...] second amendment right to own guns to defend myself'*). This author also conveys a more general conservative ideological stance (which is consistent with his stance against gun control), by placing himself in opposition to President Obama and the democrats. This conversative stance is further underlined by his chosen user handle.

ConservativeKen's comment generated two comments in direct response. The first of these (C_2) written by *dj_safari*, is pretty evidently in favor of the proposals for greater gun control, with language that overtly states this stance (*'[...] better controls on who can purchase guns [...] it's just common sense'*). This comment also includes language directed at the previous author that does not refer specifically to the topic of gun legislation, but instead expresses antipathy towards the previous author (*'[...] paranoid*

[2]Although this is a constructed example, the language and tone in these examples is consistent with that found in real comments from actual discussions on this topic. The snippet exemplifies many of the features that are pertinent for this research, and compresses them into a small discourse unit. At points throughout this thesis I will refer back to the dialog in Figure 1.1 to ground the reader with a concrete example of the concepts being discussed.

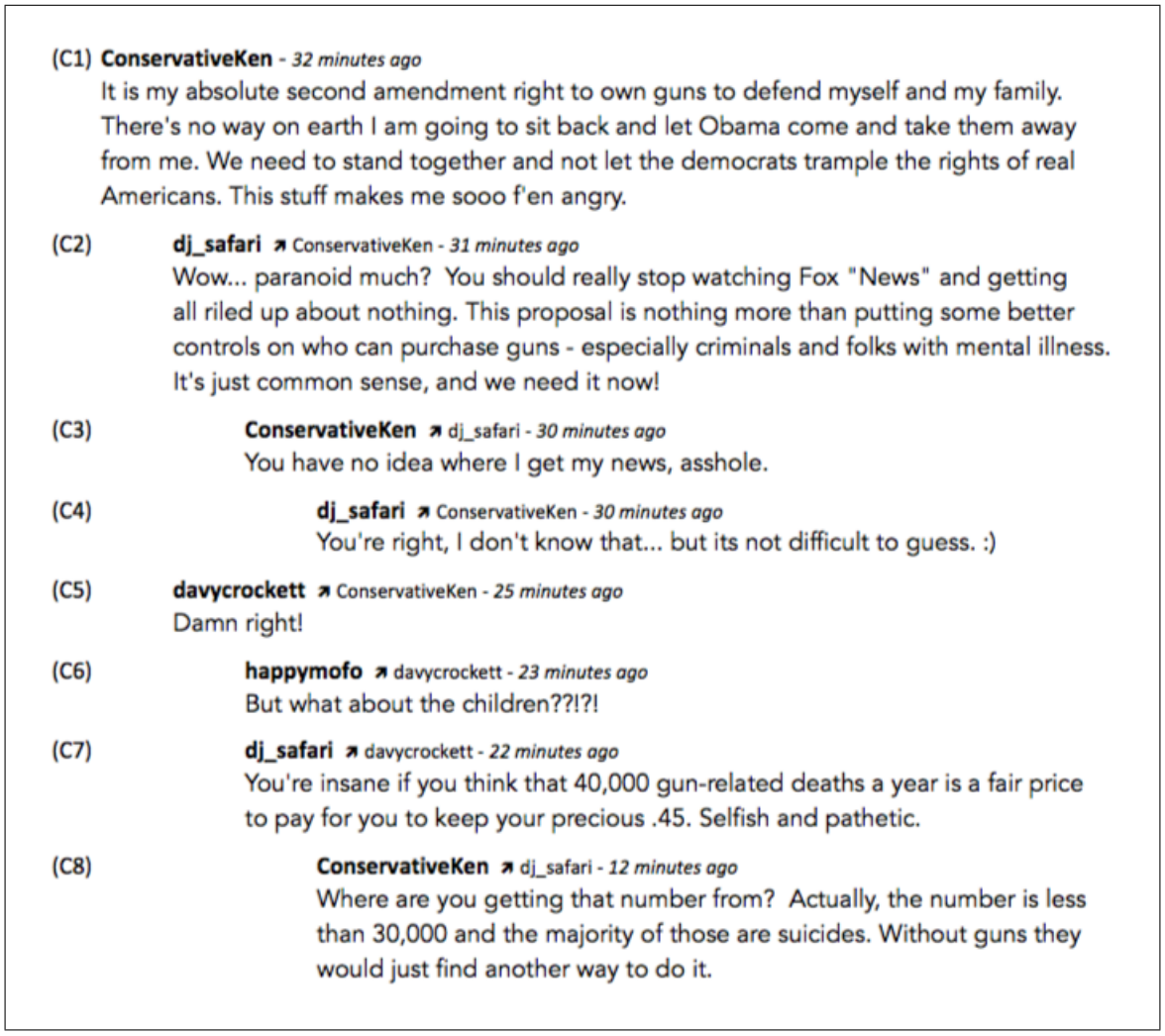

Figure 1.1: Illustrative example of an online discussion on the topic of gun control

much? You should really stop watching Fox News [...]'). This kind of inter-personal bickering between discussion participants who are in disagreement is extremely common in this genre of online discourse.

The second response (C_5) to the initial comment is by a third author, *davycrockett*, who gives no direct evidence of his stance on the topic of gun control, but instead expresses agreement with - and therefore endorsement of - *ConservativeKen*'s position (*'Damn right!'*). As a result, we readers are easily able to infer *davycrockett*'s negative topic stance. Note, however, that if this comment had been decontextualized from the discourse, with no way of relating it back to the comment to which it was responding, it would be impossible to decide the stance of *davycrockett* on gun control. The inference is only made possible indirectly as a result of the strong expression of agreement between the comments C_1 and C_5, and the overt direct expression of a negative stance in the former.

The overt antagonism between *ConservativeKen* and *dj_safari* started in C_2 picks up and intensifies with the two later comments C_3 and C_4, in which their exchanges now no longer relate to the gun control topic, but instead are simply insults (*'You have no idea [...] asshole.'*)[3] and argumentation moves (*'[...] I don't know that but its not difficult to guess.'*). Note also that the timestamp metadata on the thread of comments running from C_2 to C_4 indicates that these two commenters were essentially communicating synchronously in a real-time conversation flurry, with barely any time delay between comments and responses. As in the case of *davycrockett*'s comment C_5, the text of comments C_3 and C_4 in isolation would be insufficient for a reader to be able to deduce the topic stance of these comments' authors. This inference can only be done in the context of the chain of disagreements/opposition between successive comment pairs, and the anchor of the known negative stance of comment C_1. Human readers are able to compute this inference easily and effortlessly, but it gives a clue as to the complexity of the automatic stance detection task, and the kinds of things that a model will need to be trained to pay attention to.

[3]Owing to the (pseudo-)anonymity of internet discussion communities, disagreement between posts can often descend into insults or hostile remarks directed at other commenters, ad hominem attacks, and hate speech towards an entire class of individuals. A raft of recent work (among others: Spertus (1997), Razavi et al. (2010), Xu and Zhu (2010), Warner and Hirschberg (2012), Lukin and Walker (2013)) seeks to automatically detect such offensive comments, so to reduce the burden of human moderation of the comments posted on reputable websites. Such classification is beyond the scope of this dissertation.

One additional corner of the discussion where these two commenters interact is with *ConservativeKen*'s comment C_8 written in response to C_7, by *dj_safari*. In this case, the opposition of their respective stances is still apparent, with *ConservativeKen* questioning (*'Where are you getting that number from?'*) and then correcting (*'Actually, the number is...'*) an assertion made by *dj_safari* in the previous comment.

We can see that *dj_safari*'s comment C_7 in response to *davycrockett*, is similar to his earlier comment C_2 (in response to *ConservativeKen*) in that it contains both topic-specific language (*'... if you think that 40,000 gun-related deaths is a fair price to pay...'*) as well as direct address to the previous commenter using second-person pronouns and insults (*'You're insane ... Selfish and pathetic'*). This comment corroborates the evidence we have already collected in the matter of *dj_safari*'s (positive) stance on gun control, both with his directly-expressed opinion on the topic, as well as his hostility towards another commenter whose negative stance had already been inferred.

Finally, we have opaque and cryptic comment C_6, written by a fourth author *happymofo* (*'But what about the children??!??'*). On one hand, we might interpret this literally as a genuine information-seeking question, especially given the use of the initial contrasting connective *but*, reading it as being in opposition to the prior comment, showing the commenter's concern about young victims of gun crime, such as at Sandy Hook elementary school. With this reading, it could be interpreted as an expression of a positive stance in favor of increased gun control legislation. On the other hand, the non-standard, expressive use of question and exclamation marks might tip us off that this comment is not intended to be read literally, but instead supposed to be interpreted through an ironic filter. Under this reading, *happymofo* is performatively and mockingly adopting the tone of someone who might earnestly express such a concern, and is thereby actually expressing an anti-gun control stance. It is simply not possible to ascertain *happymofo*'s topic stance given only the context of the discussion shown in Figure 1.1; instead we must look elsewhere in the broader discourse, hoping to find an instance or two of this commenter expressing his or her opinion more directly. Alternatively, we can hope to find an example or two of a comment indicating unambiguous and unironic agreement (or disagreement) with another discourse participant whose topic stance we are relatively confident that we know.

Unfortunately, the type of playful or sarcastic language use exemplified in C_6 is rife in the genre on online discussions. Furthermore, there are frequently comments in a discourse that are off-topic entirely or otherwise uninterpretable vis-a-vis the author's stance on the topic under discussion. Unless relevant evidence can be gleaned from other sources - such as the other comments left by the same writer in the discussion (if there are any), or the commenter's choice of username (in the event that this contains a clue to an underlying ideological orientation) - we may, even as human readers, end up not being able to assign a topic stance to every single discourse participant. This will also constrain the expected ceiling performance of a stance detection classifier.

From the extended example above, we get a sense of the complexity of the author stance detection task. A naive approach might be to attempt to classify the topic stance of an individual commenter using only the direct evidence available from the decontextualized text of the posts made by that author. However, given the high preponderance of posts like C_3, C_4, and C_5 that are interactions between commenters expressing agreement, opposition, and so forth, but which do not express direct opinions on the topic under discussion, such an approach would likely be unable to predict the stance for the many participants in the discourse who only have these types of interactions. Instead, as human readers, to infer the stance of a given commenter in the discussion, we often have to rely on other available evidence of that author's stance, such as the polarity of the ideological alignment between adjacent posts, or other clues from the discourse structure and the conversation metadata, to fill in the gaps that are left by the direct evidence alone. A more sophisticated model would need to account for this multiplicity of sources of evidence.

There is also the matter of the consistency of the inferences that can be drawn from these multiple sources of evidence for a given commenter's topic stance. For example, we might notice that the direct evidence available from the text of an on-topic comment is inconsistent with the evidence from other comments written by the same author elsewhere in the discussion, or with evidence from comments posted by other authors. In an idealized discourse, every comment written an author would be consistent with a single assumed topic stance. Moreover the system of all author stances would be wholly self-consistent with respect to the expressions of agreement or disagreement that

we see in the discourse between commenter pairs. However, online discussions are far from this idealized state, and contain much more noise, not least because of the presence of playful or sarcastic language as already mentioned. When confronted with such seemingly inconsistent evidence with respect to the topic stance of an individual commenter, readers have to weigh all of the available pieces of information together, and draw a conclusion based upon the preponderance of the evidence and our confidence in the quality of each source data point. A robust author stance detection model should be able to handle such contradictory inputs, adjudicate between them, and make its predictions accordingly. In this dissertation, I present such a model.

1.4 Stance detection and sentiment analysis

I include a short discussion of how the task of stance detection is similiar to – but also crucially distinct from – the closely related task of sentiment analysis. On the one hand, the goal of sentiment analysis is to assess polarity - whether a text is **positive** or **negative** - as well as the sentiment intensity, by considering whether it contains subjective or sentiment-bearing words. This could either be a holistic assessment (a movie review, say), or aspect-related (for example, whether a user review of a camera is favorably or unfavorably disposed to the camera's picture quality). Stance detection, on the other hand, predicts whether a text (or the author of the text) is **for** or **against** (or, *pro* or *con*) a particular position. Crucially, the polarity of a predicted stance will depend upon the way in which the target issue or proposition was framed. For example, if the target proposition is 'The UK should not withdraw from the EU', the predicted stance of a text will be entirely opposite to that if the target did not contain the negation. In other words, there is no natural or necessary correlation between positive sentiment words and *pro* stances, or between negative words and *con* stances.

Moreover, a text may express a stance towards a target without using any sentiment-bearing words. For instance, in a discussion about the legalization of marijuana, a text might say '*The medical effects of marijuana have not been proven. But the consequences for society are serious.*' In this case, the opinion of the author towards the issue is clearly stated, but the text itself contains only words with neutral sentiment.

The biggest challenge for stance classification, compared to sentiment analysis at least, is that the target of interest may not be the same as the target of the opinion expressed in the text, and readers have to resort to inference and world knowledge to arrive at the text's stance toward the target in question. For example, the detection of the stance of the tweet '*Hillary is the only remotely qualified person on the ticket*' towards the target 'Donald Trump' is easy for humans, given their knowledge of the personalities involved in the US presidential election race of 2016, but much harder for a system without access to an external knowledge base. All in all, stance classification is a harder problem than sentiment analysis as it is more complicated than classifying text into expressing a positive or a negative opinion.

Stance detection can also be considered as a crucial step towards the ultimate goal of the full natural language understanding of online conversations. Fully understanding the discourse and dialogic structure such discussions would be useful for a number of things, including the ability to automatically generate summaries of the arguments on both sides of an issue. It could also be used to learn the linguistic and rhetorical devices that make for successful persuasive arguments, or the expressions of disagreement, which could then be used in other dialog systems or chatbots. Furthermore, this methodology could be applied by researchers working in the areas of social psychology or political science to learn or quantify the level of ideological 'overlap' between various polarizing social or political topics, based on the similarity of the predicted stances of authors who participate in discussions across different topics.

1.5 Thesis outline

To summarize this introductory chapter, the two research questions addressed in this dissertation are:

- In an online discussion on a polarizing ideological topic, is it possible to determine the topic stance of every discourse participant (e.g. *pro* or *con*, on the issue of gun control, say) based upon the author's contributions to the discussion, and her interactions with other discourse participants?
- If it is possible to do so, what are the features that allow for this classification, and

how do these features interact with each other, particularly when they are in conflict with each other?

The roadmap to the rest of the dissertation is as follows. In Chapter 2, I describe the process by which the datasets used in this work were selected, collected and annotated, and in Chapter 3, I provide an overview of the methodologies from natural language processing and machine learning that were used in the processing of data, and the development and evaluation of the statistical models.

In Chapter 4, I unpack further some of the complexities of the author stance detection task. I lay out the desiderata for the prediction model, provide a summary of related work, and sketch the proposed solution architecture. In Chapter 5, I describe the implementation and evaluation of a local classifier that detects the topic stance of comments in online discussions, focusing on the example ideological topic of the same-sex marriage. In Chapter 6, I present an agreement classifier that detects the ideological alignment (or disagreement) between adjacent comments in online discussions. In Chapter 7, I propose two further models that help in the author stance prediction task, utilizing features from the discussion other than those related to the comment texts. These are (i) a username classifier that determines the ideological orientation of a participant given his or her choice of pseudonym and any available self-description, and (ii) a probabilistic model that predicts the likely level of agreement or disagreement between any two authors in the discussion based solely upon the patterns of interactions between this pair in the discourse structure. In Chapter 8, I present the model for author stance prediction, using the various component classifiers from the previous chapters, and I evaluate the model over two datasets. Chapter 9 offers a conclusion, summarizing the main findings and indicating directions for future work.

Chapter 2

Data

In this chapter, I describe the process for collecting and annotating the data used for the models in this dissertation. I start by providing the motivation for the choices of the source of the data, the polarizing topic selected, and the two particular datasets used in this dissertation for development and testing. I also describe the task of getting human annotations of the stances of the authors participating in the development and testing datasets, which will serve as the ground truth against which the results of the predictive models will be evaluated.

2.1 Data Collection

2.1.1 Source

Identifying a high quality source of data for analysis was not a trivial issue. There are many different news and social media sites where users can leave comments and interact with other users, expressing their views on a topic. Three broad factors influenced my ultimate decision to use data from the news website www.politico.com: (i) the quality of the comment discussions, (ii) the balance of commenters, and (iii) the availability of the data.

First, on the quality side, I found that news sites varied greatly in the quality of their discussions. On one side, I have the esteemed New York Times, whose comment sections are heavily moderated by a team at that organization. Comments are only

posted after they have been approved, or if the commenter has an established status as a quality contributor. This results in a time lag between a comment being submitted and it showing up on the website. Consequently, the resulting comments section resembles more of a curated 'letters to the editor' format rather than a dynamic discussion in which participants respond to each other's posts and interact with each other. As the inter-comment relationship is a feature that I intend to exploit in the model, data from this source was not really suitable. At the other extreme, there are news sites such as the *San Francisco Chronicle* with comments sections that are only very lightly moderated (or which are left to the community of commenters to self-regulate). Such 'Wild West'-style comment sections generally have a much lower quality of comments, with many posts using insulting language, or being off-topic. The data from www.politico.com seemed to have the appropriate mix of thoughful comments and interaction between commenters, with some healthy disagreements, but not disintegrating into screaming matches or flaming.

Furthermore, politico.com attracts a broadly centrist (or maybe a little to the left) readership. It is not a partisan site like www.breitbart.com, an internet analog of Fox News which attracts a far-right crowd. Nor is it a place like www.npr.org, which tends to attract a left-of-center readership, without too much open disagreement in the comment discussions. Again, www.politico.com has the Goldilocks effect: not too right-wing, not too left-wing, but just about right.

Finally, www.politico.com's commenting platform is powered by Disqus, a comment hosting service for news websites and other online communities. The Disqus platform is enormous - installed on over two million websites, including major news publications such as the UKs Daily Telegraph. As of May 2013, the platform had over 100 million registered user profiles, and boasted over one billion visitors per month. Disqus provides an API allowing comprehensive access to the comment data for a given news article and the associated metadata. The API also allows certain information from registered users profiles to be collected, including the username, comment history, and a personal tagline or self-description written by the user. I make use of the information from the Disqus user profiles when I develop the username classifier in Chapter 7.

2.1.2 Topic

The choice of discussion topic was guided by a number of factors. The first requirement was that the topic is sufficiently engaging to result in expansive online discussions among a large group of commenters. The topic also needs to be polarizing, inasmuch as the discussion contributors clearly take one of two clear positions, and that it would not be difficult for a human reader of the discussion to be able to sort the discourse participants into these two subgroups. In this regard, the choice of a controversial social issue such as abortion, gay marriage or marijuana legalization seems appropriate, since most people do have and express a clearly discernible overall stance on this topic i.e. generally in favor of or against it (even if they possibly have a more nuanced position about some of the underlying details). This compares to a more diffuse debate topic such as climate change or the economy, where there is not so clear of a binary division of opinions.

Ultimately, I chose to work with comment data scraped from online discussions on the topic of the legalization of marriage for same-sex couples, in the spring and early summer of 2015. At the time of the data collection, this was probably the hottest of the polarizing social topics being discussed on news and social media, as a result of the United States Supreme Court hearings in the case of Obergefell v. Hodges, which ultimately confirmed the fundamental right to marry for everyone in their landmark ruling on June 26, 2015.

The work contained in this dissertation, however, is not applicable only to online discussions on the topic of marriage equality. The methodology and models can also be applied (subject to the availability of external training data for the topic stance classifier - see Chapter 5) to other controversial, polarizing topics.

2.1.3 Data sets

I work with two main datasets in this dissertation: a development set used to experiment with the underlying modeling choices and to tune the model parameters, and a test set upon which the model will be evaluated.

The development dataset relates to the comment discussion under the lead story on www.politico.com on June 26, 2015, titled '*Justices Rattle Both Sides on Same-Sex*

Marriage'[1] reporting on oral arguments presented to the Supreme Court with respect to Obergefell v. Hodges. The data were collected on August 22, 2015. This discussion comprised a total of 6,337 comments made by 405 unique commenters (excluding a small number of comments left by users with the screen name *Guest*). For each comment, the data collected comprised a unique comment ID, the name of the author, the text of the comment, the ID of the comment it was replying to (if any), and the date and time of post. Comments ranged in length from a single sentence to sixty-seven sentences, with an average of 2.6 sentences. The average number of word tokens per comment was 29.5.

The ID of the comment being replied to allows us to identify a post's *parent* comment, and therefore reconstruct the whole tree-like structure of the discussion. In the development set, 1,347 comments (21.3%) did not indicate a parent comment ID, meaning that these were *top-level* comments were posted directly in response to the article, rather than being responses to other comments. There were also a number of comments (151, 2.4% of the total) whose parent IDs did not appear as comment IDs elsewhere in the dataset, indicating that the parent comment to which the post was in response to, had subsequently been deleted by the author (or by the site's moderator). Just over half of the comments (3,287, some 52% of the total) received one or more comments in response; the remainder did not receive a response, and therefore served as the end of that comment chain.

The level of participation by the commenters in the discourse varied widely, with a median number of comments per author of two. There were a small number of power contributors who each wrote a very large number of comments to the discussion. The greatest number of comments left by a single individual was 169. On the other end of the participation spectrum, 76 commenters (19%) just left a single comment in the discussion and another 51 (13%) left just two contributions to the discussion. Overall, there is a general Zipfian distribution of the number of comments left by each contributor: the 10% most prolific commenters in the development set account for a full 50% of the total comments in the development set, and the top 25% frequent commenters account for 78% of all comments. This pattern seems to be fairly typical across other datasets examined.

This pattern of participation has implications on the expected performance of the

[1] http://www.politico.com/story/2015/04/same-sex-marriage-arguments-supreme-court-117424

author stance prediction model. Clearly, for the more prolific commenters the model has many more data points upon which to establish its prediction of author stance, whereas for the commenters who only write one or two comments, there is much less data to base the prediction on. Human readers encounter the same potential problem with low frequency commenters: a single comment on its own may give only a mild indication of the position of that author, but without corroborating evidence from other comments made by this person, the confidence of our assumed stance for this author will be lower. From the perspective of a practical application of the author stance classifier, it may be appropriate for the model to refrain from predicting an author stance for low frequency commenters, unless these predictions come with high associated confidences. After all, members of internet commenting communities come and go all the time, and it may not be desirable to predict the stance of every user who pops up once, leaves a single comment, and disappears forever - especially if these are not high confidence predictions. Instead it is the stance predictions for the more established, higher-level users with lots of posted comments and attested interactions with other users that will be more valuable. In Chapter 8, I show the model predictions both for the whole set of authors in the discussion, as well as broken down into high, medium, and low frequency commenters.

The test data set relates to the comment discussion under a related story on www.politico.com on June 28, 2015, titled '*Texas AG: State workers can deny marriage licenses to gay couples*'[2] reporting on the Attorney General's announcement that officials in that state were not required to follow the Supreme Court ruling which was handed down two days earlier and which confirmed a right to marry in all states. The data were also collected on August 22, 2015. The test data set is slightly larger than the development dataset, comprising 7,755 comments by 623 commenters (excluding '*Guest*'). A similar Zipfian distribution of the number of comments and commenters is exhibited in these data. Table 2.1 shows the relevant summary statistics for the development and test sets.

As a side note: the test data set was originally intended to be the comment discussion under the lead story on politico.com on June 26, 2015 [3] reporting on the US Supreme Court's historic ruling that day. This data set comprised almost 20,000 comments by just over 1,000 unique authors. However, inspecting the data it was quickly noticed that there

[2]http://www.politico.com/story/2015/06/texas-attorney-general-gay-marriage-119518
[3]http://www.politico.com/story/2015/06/supreme-court-gay-marriage-119462

Table 2.1: Summary of development and test datasets

	Development	Test
Comments:		
- Top-level	1,347	1,752
- Orphan	151	224
- Replies	4,839	5,779
Total comments	6,337	7,755
Authors:		
- Unique authors (excl. *Guest*):	405	623
- Author pairs	1,598	2,060

was an extremely high proportion of low quality comments in this data, including spam (for instance, one user had posted the very same comment over 3,000 times) as well as junk or offensive posts. The identification of low quality comments is beyond the scope of this dissertation work; the model is not readily equipped to deal with such data. Accordingly, this test data set was rejected, and the smaller dataset from a related story two days later, after the initial outburst of emotional opinion to the breaking news had toned down a little, was used instead as this did not appear to have the same proportion of low quality comments.

2.2 Gold Annotation of Author Stances

2.2.1 Task

In this section, I describe the process by which I collected human annotations of the stances of the authors participating in the development and test datasets. Recall that these judgments serve as the ground truth against which the results of the predictive models are evaluated. The goal of this annotation task was not to determine the actual topic stance for every single comment in the discussion, but instead to determine a single overall stance for each commenter based on the totality of their contributions to the discussion.

With this goal in mind, I extracted for each author a report of their contribution to the discussion, showing the username, and then listing chronologically every comment

written by that author. For an author with more than 30 comments in a single discussion, I randomly sampled 30 of her contributions. Each comment was shown in context - meaning that if the comment was written in response to a previous post, the author's name and the text of that previous comment was included.

Three human annotators (other graduate students, research assistants or friends) were asked to read carefully each of these author reports, and then to answer the question: what is this author's position on the topic of same-sex marriage? Is this commenter: (i) *pro* marriage equality (even if the commenter seems to express a general distaste for the gay community); (ii) *con* marriage equality (even if the commenter seem to be supportive of other gay rights); or (iii) is it not possible to say (i.e. there is not enough evidence from the comment texts to be able to decide conclusively)? For each author, the judge was asked to provide a rating of 'P', 'C' or 'N', respectively.

The judges were told that their decision should be taken based on the aggregate of all comments made by the commenter, and how the commenter seems to interact with other participants in the discussion. (There is an inherent assumption here that commenters have consistent views across all of their comments in a discussion, but I do not think this is unreasonable.)

It was reported that the judges found this task pretty straightforward, and even somewhat fun (given the colorful language used in some of the comments in the datasets). In the next section, I describe the results of this annotation task.

2.2.2 Results

To obtain the ground truth of the stance for each author in the development and test sets, I took the majority label given by the three human annotators. For instance, if two of the three human annotators indicated that a particular commenter had a 'P' stance on the topic of marriage equality, and the third annotator had indicated 'N' (since they were not 100% certain for some reason), the assumed actual stance was determined to be 'P'. In the few cases where the three judges all gave distinct judgments (i.e. one judgment each of 'P', 'C', and 'N'), the actual stance was taken to be 'N'. This happened very rarely, and only for commenters with a low number of contributions to the discussion. A summary of the stances for the two datasets is shown in Table 2.2. The overall ratio of

Table 2.2: Summary of human annotation of stances

Stance	Dev Set	Test Set
Pro	204	407
Con	129	127
Can't say	72	89
Total authors	405	623

pro to *con* stances reflects the slight liberal bias of the readership of www.politico.com.

The level of agreement between judges on this task was very high, especially for the commenters who made more than two comments in the discussion. Indeed, for the top 50% of most prolific authors in both discussions, there was not a single example where two annotators gave opposite judgments (i.e. one judge gave a 'P' whereas another gave a 'C'). The only disagreements were where one or two judges gave a definite opinion on a 'P' or 'C' stance, and the other judge(s) returned a 'N'. The levels of inter-annotator agreement, as measured using Cohen's kappa were 74% and 68% for the development and test sets, respectively, which indicates reasonably high levels of agreement. Based just on the top 50% of commenters by frequency, these scores rise to 85% and 71%, respectively, indicating very high levels of agreement.

Since the development and test sets were both drawn from www.politico.com, the contributors to both discussions were part of the larger community of Politico commenters, and indeed a total of 96 commenters from the April 28 discussion (development set) also contributed to the June 28 discussion (test set). For 82 of these 96 commenters, the majority judgment given by the human annotators agreed between the two discussions (60 authors were deemed to be liberal in both datasets, and 22 were consistently judged to be conservative). The remaining 14 cases related to commenters who had a majority judgment of 'N' the first dataset. In 11 of these instances, judges were able to infer a liberal or conservative stance in the subsequent dataset; in the remaining three cases, the human judges still could not tell the true underlying stance. Crucially, there were no cases at all in which the majority judgment flipped from 'P' in one dataset to 'C' in the other, or vice versa. The fact that the human judges were able to consistently determine the stance of an author given the totality of her comments to an online discussion means that this task is a reasonable one to try to automate.

My analysis of the 'N' cases showed that these occurred for two main reasons. First, there are cases where there is simply no (or not enough) evidence to be able to make a determination of stance with any confidence. A particular commenter may have left just one or two comments, which do not contain a strong indication of the topic stance. For example, she may have left an oblique or off-topic comment, or instead her post expressed alignment or disagreement with a previous comment, but not directly on the exact topic of marriage equality. The second reason is that the aggregate of the posts left by a single commenter contain conflicting signals, with some comments seeming to be in support of marriage equality, and others against it, and as a result the author seems to hold opposite positions within the same discussion. It is possible that the commenter evolved her position over the course of the discussion. However, this is unlikely, given the way people tend to become more entrenched in their positions as discussions persist, rather than change their minds over time. A survey carried out by the Pew Research Center (2016) found that only 6% of respondents said that they had changed their view on the issue of gun control based upon something – either a news story or user-generated comments – they had seen posted on social media. For the issues of gay rights and immigration, the corresponding percentages were 3% and 2%, respectively. Since the model assumes a constant topic stance for each author (i.e. does not track the course of a commenter's stance as it changes over time) such cases should be carved out and the model predictions not evaluated for them.

A second, more likely, reason is one of the author using irony - adopting the language or arguments by someone on the other side of the debate for rhetorical or humorous effect. Of course, if a user adopts the same ironic tone through the entirety of her contributions to the discussion, it will be difficult - even for a human - to spot this and determine that the author's actual topic stance is contrary to what is indicated at face value. Similarly, given enough comments made by a user, with only a few of them being ironic, human judges should be able to detect the true underlying topic stance. However, many of the 'N' cases in the development and test sets appear to be cases where the ratio between authentic and ironic comments is more balanced, leaving two or more of the judges genuinely uncertain of the author's true opinion on the topic. Again, these cases should be omitted when evaluating the results of the author stance prediction model.

Chapter 3

Methods

In this short chapter, I describe the natural language processing techniques that were applied as basic data preprocessing steps to convert the raw text of user comments into a form that can be used for subsequent analysis and modeling. I also explain some of the basic concepts of machine learning, namely feature representations, classification algorithms, and evaluation methodology.

All of the computational work in this dissertation was carried out using the Python programming language, using the spaCy library (Honnibal and Johnson (2015)) for NLP preprocessing, and various scikit-learn libraries (Pedregosa et al. (2011)) for the machine learning modeling.

3.1 Data Preprocessing

3.1.1 Normalization

I applied a process of cleaning and normalizing the raw comment text before applying the standard NLP preprocessing steps.[1] Given the provenance of the data (i.e. user generated content from the web), this will not be in as good as shape as more highly edited monologic text. The data normalization process included:

- stripping out any html markup language (i.e. removing tags such as $< b > and <$

[1]The text cleaning and normalization processes described in this chapter were carried out programmatically, and manual spot checks of the resulting texts were carried out to ensure quality.

/b >)

- replacing any text strings that look like urls with the string 'URL'
- replacing any text strings that look like email addresses with the string 'EMAIL'
- replacing any text strings that look like dollar amounts, percentages, and numerals with the strings 'DOLLAR', 'PERCENT' and 'NUM', respectively
- replacing any text strings that look like references to bible verses or chapters with the strings 'BIBLE'

I also applied a couple of text normalization steps to capture some specific characteristics of the genre of internet language. First, I replaced any text strings that resembled emoticons - such as :), :(, etc - with the strings 'POS_EMOTICON' and 'NEG_EMOTICON'.

Next, in this genre it is not uncommon for writers to use what has been referred to as 'expressive lengthening', whereby the sentiment or emotion of an author is expressed by a non-standard repetition of characters in words ('*sweeeeeet!*', '*oh noooooo!*'). To capture this, I used regular expression matching to replace sequences of three or more repeated characters in a word with just two of the same character. Another common practice in this genre is to use repeated punctuation to express emotion (e.g. '*great!!!!!!!!!*', '*really?????!!??*'). For these cases, I collapsed all combinations of repeated exclamation points and question marks into one of three possible outcomes ('!!', '??', and '?!'). I replaced all cases of ellipsis (that is, a sequence of three or more periods) with the single unicode ellipsis character.

Finally, text units separated by whitespace (i.e. a heuristic proxy for words) were lowercased before passing on to the NLP preprocessing step. The exception to this was any word that was written entirely in uppercase, since the use of 'all caps' to emphasize words may carry meaning that is useful in the classifiers.

3.1.2 Natural Language Processing

In this dissertation, I utilize the outputs produced by basic NLP text pre-processing methodologies, in particular: sentence and word tokenization, part-of-speech tagging,

lemmatization and dependency parsing. For completeness, I describe briefly each of these terms in turn.

Tokenization is the task of splitting running text into pieces of text called tokens. A token is an instance of a sequence of characters that are grouped together as a useful orthographic unit for processing. Tokenization is a relatively easy task, in English at least, in which words are separated by white spaces and punctuation. Sentence tokenization is the related task of splitting sequences of tokens into sentences.

Part-of-speech (POS) tagging is the process of marking up the tokens identified in text as corresponding to a particular part of speech, based on the context in which it is used. POS tagging provides both coarse-grained and fine-grained tags for each word token. The coarse-grained POS tags represent the basic word classes (noun, verb, adjective, etc), whereas the fine-grained POS tags also include morphological information such as verbal tense or aspectual information for verbs (such as 'VBD' and 'VBG' for the past tense and present participle forms of verbs, respectively), and number for nouns. The state-of-the-art systems for POS tagging report accuracies above 97%.

Lemmatization is the process of converting the word tokens in a text into their dictionary form. To be precise, this means that nouns drop any plural marking, and finite verbs are changed to their uninflected form, losing information about tense, aspect and number.

Dependency parsing is a method of analyzing the grammatical structure of a sentence, establishing relationships between word tokens in a sentence as a tree structure. In the dependency tree, each word token is represented as a node, and each edge between nodes is labeled with a dependency relation from a given set. Such relations include *nsubj* (between a subject and its head verb), *dobj* (between a direct object and its verb), *det* (between a determiner and the head noun it is modifying), and so on. Dependency parsing is an alternative to the more traditional syntactic constituency parsing, and has many applications in language processing. This way of parsing a sentence very naturally shows the head-argument relationships between lexical items, and the resulting representation is flatter than that given by full phrase structure parse.

I utilize an open source Python library called spaCy (Honnibal and Johnson (2015)) for all of these NLP pre-processing tasks. spaCy is a library for industrial-strength nat-

ural language processing in Python, and provides state-of-the-art speed and accuracy. Moreover it is trained to attempt to handle messy data, including emoticons and other web-based features. Given an input text document, spaCy carries out tokenization, sentence recognition, part of speech tagging, lemmatization and dependency parsing in a single step.

3.2 Machine Learning

In this section, I discuss some machine learning basics, including feature representation, binary classification modeling, and evaluation. I also describe two external sentiment lexicons used in the model.

3.2.1 Feature representation

I use the standard n-gram bag-of-words (BOW) feature representation for the classifiers presented in Chapter 5 and Chapter 6. An n-gram is a contiguous sequence of n tokens from the span of text. In the cases of n equal to 1, 2 and 3, the corresponding terms used are unigrams, bigrams and trigrams, respectively.

n-grams are converted to features using the method of 'one-hot' encoding. To begin with, there is a first pass through the entire training set to determine the overall corpus vocabulary. For this stage, I can choose to ignore n-grams that occur less than a minimum number of times in the training set, as well as highly frequent unigrams that occur in a stoplist of basic function words, such as *the*, *an*, *of* and so on. In this work, I use the basic list of 132 stopwords included with the Natural Language Toolkit (Bird (2006)). Then, each data instance is converted to a feature vector of length M (the number of retained n-grams in the vocabulary), where the m-th entry of the feature vector is set equal to 1 if the instance contains the m-th term in the vocabulary, and 0 otherwise. The resulting features are very sparse, meaning that the vast majority of entries will be zero for each data instance, since a given text will only contain a very small subset of the overall number of n-grams in the corpus.

It is called a bag-of-words representation because the order in which the n-grams appeared in the original text is not preserved (other than the very local word relation-

ships that are captured within each n-gram). One may think that losing the information about the linear order of the words in a text would be deleterious in the task of classifying the text, given how much meaning depends on the syntactic structure of the sentence. However, despite not capturing hierachical or longer-distance dependencies, basic n-gram representations tend to do surprisingly well for many text classification tasks, often performing as well as, or better than, more sophisticated linguistically-informed features that attempt to recover syntactic structure. Almost all text classification models make use of n-grams as features for model training, if for nothing else than as for a quick baseline against which more sophisticated feature sets can be evaluated.

3.2.2 Classifiers

In this work I experiment with two supervised binary classifiers, namely Logistic Regression and Support Vector Machines (SVMs). These models are *supervised* in that they require a training set of instances with known output labels. A supervised classifier learns a mapping from the set of input feature vectors to the output labels. Once the model has been trained, it can be applied to previously unseen data instances that do not have known output labels to make predictions. Logistic Regression is a *discriminative* model in that it only cares about estimating the conditional probability distribution $P(y|\bar{x})$ of the output label y based on a vector of one or more predictor variables, $\bar{x}$. In a similar way, SVM is discriminative because it explicitly learns the class boundary between the two classes. Discriminative models are contrasted to *generative* models, that seek to learn the joint probability distribution of the input and output variables, $P(\bar{x}, y)$.

Specifically, Logistic Regression is a classifier that estimates the probability of a binary response based on one or more predictor variables ('features'). It uses the logistic function to convert the linear combination of feature weights and feature values into a real number lying between 0 and 1, from which the categorical class prediction can be deduced. The features weights are learned via maximum likelihood estimation, and are directly interpretable as the increase in log odds (i.e. the ratio of the probability of a positive to a negative class label) given an increase of one unit of the feature.

An SVM is a 'maximum-margin' classifier that finds a linear boundary in the feature space that optimally separates the positive and negative instances, and uses this

decision boundary to classify unseen instances. SVMs learn a decision function f from the set of positive and negative training instances such that an unlabeled instance x is labeled as positive if $f(x) > 0$. This function f represents the hyperplane that separates the positive and negative instances.

Both Logistic Regression and Support Vector Machines have been widely used in text classification tasks, and both learners can also output an associated confidence of the predicted class label for each data instance, which is a critical ingredient of the overall author stance prediction model presented in Chapter 8. As I will explain in detail in that chapter, the associated probabilities of the predictions coming out of the underlying topic stance and agreement classifiers feed directly into the loss function for that model, and provide a means by which potentially inconsistent information from the component classifiers is adjudicated. For Logistic Regression, the predicted probability of the positive class is in fact the fundamental underlying output of the model. This probability is then rounded up or down respectively to arrive at the predicted binary class label. On the other hand, the fundamental output of an SVM is the predicted class label for a given data instance. The confidence associated with that prediction is estimated indirectly and depends upon the distance of the data point away from the separating hyperplane. A fuller description of the calculation of the predicted class probabilities in SVMs is given in Platt et al. (1999).

There are other binary classifiers in the machine learning toolbox, but these were not suitable for this work, for one reason or another. First, decision trees and random forest models can achieve high classification accuracy, but they do not easily have a way of obtaining the confidence of the predicted class labels that is necessary for the downstream task of author stance prediction. A different model, Naive Bayes, does yield the desired probability alongside its binary predictions of class categories, however this is often not a meaningful indication of the confidence of the model if the underlying assumption of the independence between the predictor variables does not hold. Finally, modern-day deep learning, neural network models would require vast amounts more training data that is available in the datasets considered in this dissertation. Moreover, they produce models that are essentially uninterpretable, and do not give any insight into the data. I do not consider these types of models in this dissertation.

3.2.3 Evaluation

The way that machine learning models are evaluated is to compare their predictions to a test dataset for which the true class labels are known. This usually means setting aside some of the original labeled data to form a held-out set, and keeping the model completely blind to this evaluation set when building the model and fine-tuning the features and parameters. The model should only be applied to the held-out test set at the very end. This provides a check that there is no overfitting – a phenomenon where the model becomes overly committed to the characteristics of the dataset on which it was trained, and therefore less generalizable to data instances that it has not seen before in the training stage. For both the comment topic stance classifier and the agreement classifier in Chapters 5 and 6, respectively, 20% of the original data set was randomly selected for testing and excluded from any training and development.

Within the training set, it is often useful to hold out a further 'validation' set against which to experiment and compare the results of different models or combinations of features. For the purposes of this work, I carry out five fold cross-validation, meaning that the training set is randomly divided into five subsets, and for each cut, a model is trained on four-fifths of the data, and evaluated against the other one-fifth. After the best model and feature set has been found via cross-validation, the final model is evaluated against the held-out test set.

Another standard way within machine learning to overcome potential overfitting is to introduce **regularization**. Regularization is a means by which more elaborate models with more parameters and higher feature weights are penalized with respect to simpler models with fewer parameters. With regularization, a cost is added to the classifier's loss function for each additional parameter in the model. At training time, the model will find the set of parameter values that minimize the loss function (that is, it will seek the find the parameters that make the predictions of the model as close as possible to the actual known class labels of the training data). The model will need to find the right balance between having a low loss by adhering very closely to the actual class labels (at the risk of overfitting the training data) but incurring a higher regularization penalty, and incurring a higher loss by fitting to the training data less rigidly but having a lower regularization penalty (and likely a more generalizable model). The balance between the

loss and the regularization penalty is controlled by the type of regularization used, and the weight given to it. These hyper-parameters in the model are usually determined by (cross) validation on the training set.

There are a number of possible metrics used to evaluate the performance of a binary classifier. The most straightforward is **classification accuracy**, which measures the proportion of correctly classified examples in the test set. However, if there is not a balanced distribution of classes in the test set, classification accuracy alone may give a misleading impression of the performance of the classifier. This situation would almost certainly be the case for the agreement classifier presented in Chapter 6. Given the generally adversarial nature of interactions between participants in online discussions, we would be able to achieve high classification accuracy using a model that simply predicts a label of disagreement to every comment-response pair, without any regard at all to the text of either comment. To assess how good a model truly is at distinguishing between cases of agreement and disagreement, we can instead calculate other evaluation metrics, such as the classifier's **precision** and **recall** for each class, and from these determine the overall **F1 measure**. These are defined as follows:

Precision is defined as the proportion of the cases predicted by the model to belong to a particular category that turn out to have been classified correctly. In other words, how accurate are the models predictions for those cases it predicts to be positive? For example, if a model predicts that 120 of the instances in the test set are positive, and, of these, 90 were indeed true positive cases, then the precision score is $90/120 = 0.75$.

Recall is defined as the proportion of the total cases in the test corpus belonging to a particular category that were correctly classified by the model. In other words, how good is the model at finding all of the positive cases in the test set? For example, if there are actually 150 positive examples in the test set, and the model corrected predicts 90 of these cases, then the recall score is $90/150 = 0.6$.

The **F1 measure** combines the precision and the recall values to provide a single score between 0 and 1. It is a weighted average (specifically, the harmonic mean) of the two values. The F1 measure can be interpreted as smoothing out large disparities between the precision and recall measures, and the resulting single evaluation metric allows for two competing models to be compared easily. For the above example, the F1 measure is

calculated as: $2 * 0.75 * 0.6/(0.75 + 0.6) = 0.9/1.35 = 0.67$.

3.3 Sentiment Analyzers

This dissertation work makes use of two external resources for sentiment analysis: (i) the MPQA Subjectivity Lexicon (Wilson et al. (2005)), and (ii) SentiStrength (Thelwall et al. (2012)). These are described in the following paragraphs.

MPQA Subjectivity Lexicon

The MPQA Subjectivity Lexicon is a resource comprising 2,718 positive and 4,912 negative terms drawn from a combination of sources, including the General Inquirer lists and a bootstrapped list of subjective words and phrases, that was then hand-labeled for sentiment. Each phrase in the lexicon is also labeled for reliability (strongly subjective or weakly subjective). The numerical sentiment scores range from -2 (highly negative sentiment) to +2 (highly positive sentiment).

SentiStrength

SentiStrength is an open source application that estimates the strength of positive and negative sentiment in short texts. For a given text (typically, a sentence), SentiStrength returns both a positive and a negative sentiment score, both scores in the range from 1 to 5. The algorithm relies on an external lexicon to find the sentiment of individual words in the text, and then adjusts these to allow for booster words, intensifiers, and the scope of any negation particles in the text. In this way, a sentence such as *"I really love you, but dislike your cold sister"* would receive a positive sentiment score of 4 and a negative sentiment score of 3. A full description of the SentiStrength algorithm is provided in Thelwall et al. (2012).

Part II

Author Stance Prediction

Chapter 4

Model Overview

4.1 Introduction

In this chapter, I start addressing the research question of whether it is possible to automatically detect the topic stance of authors participating in an online discussion on a two-sided ideological debate. I begin by picking up the thread from the introductory chapter of this thesis, in which I walked through a prototypical example of a discussion found in online forums, and highlighted a number of features of this discourse genre that present challenges for the task of author stance detection. In this chapter, I unpack and elaborate on these issues, discussing the implications that this has for the desiderata of an author stance detection model. I will hone the intuitions for the model and then sketch the proposed system architecture that meets these criteria. (The full specification of this model is deferred until Chapter 8.) I will finish the chapter with a review of related work in this area.

4.2 Model desiderata

In Chapter 1, I showed that the main challenges facing an author stance model related to the following two broad issues: the **multiplicity of sources of evidence** for an author's stance throughout a discussion, and the **potential for inconsistency** between these difference sources of evidence. Both of these aspects have major implications for how the model should be designed.

4.2.1 Multiple sources of evidence

The text of a comment is sometimes sufficent on its own for us to be able to discern the topic stance of its author, since the comment is squarely on-topic and the writer's attitude is conveyed directly. For instance, in the example discussion in Figure 1.1 in Chapter 1, the stance of author *dj_safari* can be inferred easily from the text of comment C_2 ('*This proposal is nothing more than putting some better controls on who can purchase guns - especially criminals and folks with mental illness. It's just common sense, and we need it now!*'). If all comments in a discussion were as explicit as this, it would not be too difficult (with a possible wrinkle presented by pronoun reference) to train a model that is able to classify instances of comments as either for or against gun control, with high accuracy.

However, as we saw in the introduction, the comment text of a post taken out of its discourse context is often insufficient to be able to infer the ideological position of the post's author, either because it addresses the topic but does not make the writer's attitude towards it clear, or because it does not contain any topic-specific language at all. For example, in the example discussion we cannot determine the stance of the author *davycrockett* towards gun control given solely the text of comment C_5 ('*Damn right!*'). Instead, the only information this comment gives us is a strong indication that the commenter is in agreement with his interlocutor. If we were able to determine the stance of the author who wrote the prior post, then we would be able to indirectly infer *davycrockett*'s stance as being the same as this. So the polarity of the alignment (i.e. the agreement or disagreement) between two adjacent posts is an alternative source of evidence available with respect to the stance of a discourse participant, that can fill in the gaps that would exist if we only looked for direct indicators of topic stance in the text of the comment.

In short, some comments, like C_2, may contain only topic-specific language that are interpretable without the context of the preceding discourse, whereas others like C_5, require the discourse context in order to be interpreted correctly.[1] We need both of these 'views' of the data in order to get a more complete picture. The direct indicators of

[1] And some comments, such as C_7 are a hybrid in that they contain some elements that are interpretable without the context of the parent comment ('*You're insane if you think that 40,000 gun-related deaths a year is a fair price to pay for you to keep your precious .45.*'), but need this prior context for other elements ('*Selfish and pathetic.*').

topic stance available from a comment text, and the indirect evidence that comes from looking at the polarity of the alignment between a comment and its response are in a sense orthogonal to each other, in that the presence of one type of evidence is not dependent on the presence of the other type of indicator. This suggests that the overall author stance classifier should be designed to include separate components that are able to detect each of these types of evidence independently.

The first model component should look for language in the comment text itself that directly indicates the author's topic stance. Such language could take many forms, including an explicit assertion of support or opposition to the topic, an expression of alignment with the ideological or political left (or right), an appeal to a specific argument on one side of the debate or the other, the use of sentiment-laden terms directed at topic-specific targets, the particular choice of vocabulary items, and so on. Accordingly, it would be necessary to develop a **comment topic stance classifier** that takes as input a comment text isolated from the discourse context, and outputs the prediction of whether the comment is for or against the topic stance.

The second component should look for indications of alignment or disagreement between a comment and the previous post in the discourse. These indicators can include explicit expressions of agreement or disagreement. Alternatively, this alignment can be expressed more obliquely: on the one hand, by means of praise or thanks, or, on the other hand, by insults, corrections, or questioning an interlocutor's assumptions, etc. Note that the unit of analysis for this second model component should be a comment-response pair, rather than the decontextualized comment text that is analyzed by the first component. The comment-response pair will be the input to an **agreement classifier** that outputs the prediction of ideological alignment between two adjacent posts.

There are other views of the data that can also be leveraged for the author stance detection task. For example, as we saw in the example discussion, one author *ConservativeKen* has a chosen username that strongly implies a stance on various social and political issues, such as gun control. This information is independent of the evidence that is available from the texts of his posts. Even if this author had left no on-topic comments in the discussion, and his interactions with other commenters did not provide a great deal of evidence of his stance on gun control, we may still be able to infer his topic stance

given his choice of username. Consequently, we want the author stance model to include another component that scans the discussion solely to extract the author usernames, and then classifies them for their ideological orientation.

Finally, another view of the data is given by the discourse structure itself - namely, the patterns of interaction between participants (i.e who interacts with whom, and how often) - with the actual comment texts ignored entirely. Given the genererally disputative nature of online discourse, the intuition here is that the greater the level of back-and-forth interaction between a given pair of dialog participants, the greater the likelihood is that those two individuals are engaged in an argument and so by extension can be assumed to disagree with each other ideologically. Consequently, it would be beneficial if the author stance detection model could be tuned to pick up on this evidence also.

In summary, the multiplicity of the potential sources of evidence of an author's stance suggests that the author stance prediction model should include four component models that are designed to take different perspectives on the data, and are trained to detect different features that could indicate the author stance (namely, comment topic stance, agreement between adjacent comments, username features, and indicators from the discourse structure). The proposed system architecture sketched in Section 4.3 shows how these component pieces are integrated.

4.2.2 Consistency of evidence

In an ideal scenario, the various pieces of evidence supporting the stance of an author throughout a discourse are self-consistent and align perfectly with the author's true underlying topic stance (*pro*, say). This would mean that any individual comments containing topic-specific language reflect the same *pro* stance, and any interactions with known *con* stance authors show indications of disagreement (or, at least, they do not show indications of agreement). Other indications of stance from the username and discourse structure are also expected to be consistent with the *pro* stance. We see this in the example discussion from Chapter 1 for the author *ConservativeKen*. This discourse participant unambiguously expresses his stance against gun control using topic-specific language (comments C_1 and C_8), and bluntly expresses non-alignment with *dj_safari* (comments C_3 and C_8) - who is clearly in the *pro* gun control camp. Morever, his choice of username

is consistent with a right-leaning ideology. If we were to apply the four component models described in the previous section, we would hope to find that every prediction from these classifiers in respect of *ConservativeKen* would support and reinforce the other predictions.

Unfortunately, this beautifully self-consistent picture is not often borne out in practice. In many cases, we find that the component classifier predictions for a given author are not entirely mutually consistent. For instance, maybe the predictions of the comment topic stance classifier for two adjacent comments are both negative (indicating a *con* stance), but the agreement classifier predicts a negative agreement score, indicating that the two comments disagree in their stance on the topic. This could happen if, for example, one comment disagrees on some small detail or unrelated matter in the prior comment (and the classifier picks up on the language relating to the disagreement), but the commenters in fact fundamentally agree on the topic itself. Or it could be the case that for all of the comments posted by a particular author in a discussion, some were determined by the comment topic stance classifier to have a *pro* stance, but a few were predicted to be *con* comments. These inconsistent predictions could arise for one of a few reasons. For one, the text of a comment may not directly signal the true stance, as the author may appear to be taking the opposite stance to her actual position for rhetorical effect. As a result, the comment topic stance classifier will return an incorrect prediction. This will also be the case if the author uses sarcasm.

The more likely reason for inconsistent predictions, though, is as a result of the performance limitations of the component classifiers. Unless a training set contains positive and negative instances that are exactly linearly-separable, no classifier - no matter how many features it relies on, nor the training set size - can be expected to achieve 100% accuracy. Consequently, the noise in the component classifiers predictions will lead to misclassification errors and result in predictions that are inconsistent with other evidence.

Human readers, when confronted with seemingly inconsistent evidence with respect to the topic stance of an individual commenter, will weigh together all of the available sources of information, and come to a conclusion based upon the balance of evidence and the confidence in the quality of each source data point. The automated author stance

detection model will need to do the same thing, assessing the amount and the quality of information suggesting that an author's topic stance is one way or the other, and choose the outcome that has the greater aggregated evidence in its favor. I explain how this happens in the following section.

One final point: the desideratum to be able to use the confidence of the component classifier predictions in the downstream author stance model puts some constraints on the types of classifiers that can be used for the comment topic stance and agreement models, since some classifiers such as decision trees can provide only a predicted class *label*, but not the associated confidence of the prediction. I address this in later chapters.

4.3 Proposed system architecture

In the previous section, I provided motivation for how the author stance model needs to include component elements that each are designed to identify different features of the discourse (that is, comment topic stance, agreement between adjacent comments, username features, and indicators from the discourse structure). I also explained the need for the model to use the confidence associated with the predictions of each of these components in order to assess contradictory information, and to arrive at an overall prediction of the author's stance that is most consistent with the evidence available from the component classifiers (which are assumed to be correct). The question now is what is the best way to hook these components up to each other so that information can flow between them?

There are a number of possible ways that this could be done. One approach would be to work sequentially, starting with the 'root'-level comments written in direct response to the news article at the top of the discouse tree, and use the comment topic classifier predictions for these comments (which in theory should be easier to classify since they only contain topic-specific language, and so could more easily be interpreted without the discourse context). Then, one could iterate down the cascading chain of comments in the thread that is dominated by the root, and determine the stance of each reply comment by checking the agreement classifier's prediction of the alignment between the reply and the prior comment in the chain, and reversing or preserving the predicted

polarity accordingly. We would need to deal with cases where the comment topic stance prediction for a comment is not consistent with the assumed stance of the prior comment in the thread adjusted for the expected agreement between the comment pair. This could probably be achieved making use of the associated confidence predictions. We would also need a methodology to come to a single, overall prediction of topic stance for an author, given that some of her comments may have been classified as having a positive stance, whereas others were classified as negative. Possibly, using the majority category label may be sufficient to achieve this.

However, the crucial flaw in this model design is the privilege that is given to the 'root' level comments. A misclassification error for a post at the very top of a comment chain would be propagated downstream and potentially amplified as the algorithm works its way down the chain of discourse, adjusting its predictions for successive comments based on the predicted agreement between them. A better model should be agnostic with respect to the order that the classifier predictions are integrated, and allow for the possibility that a direct, unambiguous post further down the discourse tree, with an associated high confidence prediction from the component classifier, could be used to inform the predicted stance of a comment higher-up in the tree structure.

Another way might be to start with the results of the agreement classifier - after all, the majority of the comments posted in a discussion are replies to other comments, and many of these convey clear indicators of agreement and disagreement, so the agreement classifier predictions should be pretty robust. We could build a graph representation of the discussion, with nodes representing the comments, and the signed edges between them representing the predictions of the agreement classifier, and then use a graph cutting algorithm to find the bipartite division of comment nodes that maximizes the level of intra-cluster agreement and inter-cluster disagreement. The predictions of the comment topic classifier could then be used to assign a single stance polarity to both groups.

The major disadvantage to this line of approach is that we are not fully utilizing the predictions of the comment topic classifier in this task (or the predictions from username classifier at all), other than in the final task of assigning labels, and so would effectively be throwing away useful information. By not considering this, the model would be less able to counteract any agreement classifier misclassification errors - like those we see when

a comment pair contains language that reflects some local disagreement on a minor or off-topic matter, but where the comment authors share the same topic stance. A more principled model would not simply stipulate that the agreement classifier predictions were more reliable than any other indicators.

In summary, we prefer a model that doesn't privilege the chronological order in which the discourse contributions were made, nor assumes *a priori* that one type of evidence for an author stance should get priority over another. Instead we should strive for a model that **collectively classifies** the author stances to be, in aggregate, most consistent with the component predictions. Consequently, we will need a metric that scores the consistency of the overall model predictions of author stance to the underlying component inputs, and allows for a comparison of the level of consistency under different scenarios. To operationalize this, I develop a cost function (to be described in detail in Chapter 8) that calculates a penalty when the author stance predicted by the final model is inconsistent with the evidence given by the component classifier predictions. The magnitude of the penalty should reflect just how far off the mark the component prediction is, compared to the author stance. For example, we want a higher penalty for a mismatch between an incorrect component prediction and the author stance if the classifier was very confident in its prediction, than we would if the classifier were less confident in its incorrect prediction. A corollary to this is that we will incur a small penalty even in the case of a correct component prediction when the classifier is not strongly confident of this prediction.

In the simplest case, for a given author a_1 the total penalty incurred as a result of being inconsistent with the component predictions is calculated twice: once assuming that the author's true stance is *pro*, and then again assuming a *con* stance. The resulting prediction for the stance of this author is the polarity that generated the lower penalty. However, we cannot just assess the stance of author a_1 in isolation, since many of the component predictions arise from the agreement classifier. For the pair of authors a_1 and a_2, we need to calculate the total penalty incurred under four different scenarios (i.e. *pro/pro*, *pro/con*, *con/pro*, and *con/con*) and choose the configuration of the two stances that maximizes the consistency not only with the comment topic stance classifier predictions for posts written by both a_1 and a_2, but also with the agreement classifier predictions

for comment-response pairs involving these two authors. We can quickly extrapolate this thinking to see that, given the connectedness of the authors in the discussion, it is necessary the stances for all authors in the discussion must be jointly inferred at the same time. This affects the mathematical complexity of the inference task significantly, and in Chapter 8 I will describe the mathematical optimization technique employed to solve it.

The preceding discussion will make more sense if I show how the methodology might look for the example discussion in Chapter 1. Let's focus first on the authors *dj_safari* and *davycrockett*. The former posted a total of three comments[2][3][4] in the discussion, the latter author posted just a single comment[5], and there was once instance of a comment-response pair[6] involving the two. Now for the sake of illustration, let's assume that we have the following predictions from the component classifiers. The comment topic stance classifier predicted probabilities of 0.9, 0.5, 0.5, and 0.7 for the comments C_2, C_4, C_5, and C_7, respectively. This can be interpreted as a very high confidence of a positive stance for C_2, a slightly less confident prediction of positive stance for C_7, and effectively no prediction for C_4 and C_5 (which did not contain any topic-specific language). The agreement classifier returned a predicted agreement score of 0.2 for the comment pair (C_5, C_7), which represents a high degree of confidence that these comments disagree in their stance.

Now let's cycle through the four possibilities for the stances of *dj_safari* and *davycrockett*, and show how the working of the model matches our intuitions. For the sake of illustration, let's assume that the penalty function is the absolute deviation between the prediction and the actual stance. If *dj_safari* truly had a negative stance, the component predictions for C_2, and C_7 would be wildly wrong, incurring penalties of 0.9 and 0.7, respectively. There would also be a penalty of 0.5 incurred with respect to C_4, but as we will see, the very same penalty would be levied assuming a positive stance, so this one is awash. Similarly, there would also be a penalty of 0.5 incurred with respect to comment

[2] C_2: Wow... paranoid much? You should really stop watching Fox "News" and getting all riled up about nothing. This proposal is nothing more than putting some better controls on who can purchase guns - especially criminals and folks with mental illness. It's just common sense, and we need it now!

[3] C_4: You're right, I don't know that... but its not difficult to guess. :)

[4] C_7: You're insane if you think that 40,000 gun-related deaths a year is a fair price to pay for you to keep your precious .45. Selfish and pathetic.

[5] C_5: Damn right!

[6] $C_5 \rightarrow C_7$

C_5, no matter what the assumed stance of *davycrockett*, and so we can ignore this too. Now, if we assume that *davycrockett* has a positive stance, we would expect there to be disagreement between the two authors. This is actually consistent with the prediction of the agreement classifier in respect of the comment pair (C_5, C_7), and so we would only incur a very small incremental penalty (0.2), bringing the total to 1.8. On the other hand, if we assume *davycrockett* also has a negative stance (the same as *dj_safari*) this is highly inconsistent with the agreement classifier prediction, so we would incur a penalty of 0.8, raising the overall penalty to 2.4. Running the same math assuming a positive stance for *dj_safari*), we find lower total penalties overall, since now the predictions for C_2, and C_7 are in alignment with the assumed stances. The resulting optimal stance labels for the two authors are a *pro* stance for *dj_safari* and a *con* stance for *davycrockett*.

It is not difficult to see how the approach can be generalized to assign stances to all four authors in the example discussion. There will be total of $2^4 = 16$ possible stance combinations to check, but some of these can be eliminated quickly. Note that the agreement classifier is likely to predict a strong positive agreement between the comment pair (C_3, C_4)[7], even though the expression '*you're right*' here is not being used to agree on a substantive point relating to gun control. Let's say that the agreement classifier outputs a prediction of 0.85 for this case, thereby incurring a high penalty if the assumed stances of *ConservativeKen* and *dj_safari* were different from each other. To reduce this penalty, we could set the stances of the two authors to be equal. However, doing so will actually result in a higher overall penalty, given the multitude of other evidence which would now be inconsistent with this assumption. Effectively, the raft of consistent evidence of a *con* stance for *ConservativeKen* and a *pro* stance for *dj_safari* would overwhelm the small amount of misleading evidence that they shared a stance given by the agreement classifier's prediction for (C_3, C_4).

This discussion provides a sketch of how the collective classifier would work to jointly infer the stances of the authors in the discussion. Chapter 8 contains the details of the actual penalty function and the methodology used to find the optimal configuration of author stances that results in the lowest possible penalty.

[7] C_3: You have no idea where I get my news, asshole. $\rightarrow C_4$ You're right, I don't know that ...

4.4 Related Work

This is not the first work to attempt the task of author stance classification in online discourse using information other than or in addition to simply the text of comments taken in isolation from the discourse context. A few works attempt to address the stance classification problem by considering solely the link structure created by the comment-reply pairs in the discussion and the resulting social network structure between commenters, with no or minimal consideration given to the actual textual content of the comments. Agrawal et al. (2003) assumes all adjacent comments disagree with each other, uses no information at all from the text itself, and applies the MaxCut algorithm to partition the graph of the network of users into those taking supporting and opposing stances. This methodology is adopted and further developed by Murakami and Raymond (2010), who work with a small dataset of 481 comments in total from 175 commenters posted to a public opinions website. They incorporate a very small number of basic opinion expressions extracted from the content of the posts before applying a similar clustering algorithm in order to partition participants into supporting and opposing parties. They show that the combination of both link and simple text information leads to a small improvement in model performance.

Other works implement a more sophisticated approach to collectively classify the stance of posts in a discussion, as in this thesis. These studies rely on both the linguistic features of the posts and features that capture the underlying relationship between posts and authors, and between authors. A wide range of different modeling strategies have been proposed. Malouf and Mullen (2008) look at a dataset of 77,854 posts from 408 commenters on a political debating website, and show that a combination of textual and social network features (as derived from the co-citation matrix among the discourse participants) provides better model performance for predicting the ideology (*left* or *right*) of commenters on a political debate site than textual features alone. Their best performing model achieved classification accuracy of 68.5%.[8] Walker et al. (2012a) address the problem by creating a graph representation of the online discussion incorporating the dialogic structure of the debate, in which the nodes of the graph are the debate posts and the

[8]The performance rises to 73.0% if the analysis is restricted to commenters who posted more than 500 words in aggregate in the training corpus.

weights of the edges reflect the predicted agreement or disagreement between the posts. The resulting graph is partitioned using the MaxCut algorithm, and the orientation of each partition is determined by standard text classification methods on the aggregated text of the posts in the partition. Their model performance varied widely, depending on the topic under discussion, with the lowest classification accuracy of 33% for a debate on immigration, up to 84% for a debate on gay marriage, with an average of 65% across 14 topics. This compares to a baseline accuracy of 58%, using a decision tree classifier based solely on textual features of the debate posts.

Hasan and Ng (2013a) seek to improve upon the performance of the simple direct stance classifier presented by Anand et al. (2011), by superimposing three sets of 'soft' constraints. These are: author constraints (i.e. two posts written by the same author for the same debate domain should have the same stance); ideological constraints (i.e. the cross-domain abstraction of the author constraints); and user-interaction constraints, which seek to reflect commonly attested sequences of post stances, such as PRO-CON-PRO. The authors carry out the inference of the model parameters using the method of Integer Linear Programming. They found that including these extra-linguistic constraints increases the average classification accuracy over the four ideological topics they consider by some 11.6%, compared to the baseline accuracy of 61.8% using the text features of the debate posts alone.

A different approach is taken by Qiu et al. (2013), who present a generative, latent variable model to mine information about the interaction between participants in online debates and use this to help identify stances. Their model assumes three types of words in debate posts: (i) a topic-specific word distribution, (ii) a side-specific word distribution (which reflects that users on different sides tend to have different preferences for the usage of words, which is related to the phenomenon of 'framing'), and (iii) an interaction word distribution that reflects how users interact with each other. After an interaction feature identification stage using Gibbs sampling to mine interaction features from structured debate posts, a clustering algorithm is applied over the set of debate authors incorporating user consistency constraints in order to determine the two groups of related authors. Their best performing model achieves a classification accuracy of 62.2% averaged over 32 popular debates scraped from the www.createdebate.com website, with

an average of 170 posts by 45 authors per debate.

Finally, Sridhar et al. (2014) conceptualize the data from online discourse forums as a multi-relational network and some partially observed labels. The problem then is how to infer all of the unobserved labels, conditioned on observed attributes and links. They tackle the problem using Probabilistic Soft Logic, a framework for probabilistic modeling and collective reasoning in relational domains, to jointly classify the stances of the posts and the authors in the discussion. Posts and authors are represented as variables in the PSL model, and predicates are specified to encode different interactions between them, including unary predicates (e.g. $\textsc{isPostCon}(P)$), relations (e.g. $\textsc{disagreesAuth}(A_1, A_2)$, and rules relating them. The underlying probabilistic model is a hinge-loss Markov Random Field, and inference is a convex optimization, which leads to a significant improvement in efficiency over discrete probabilistic graphical models like the one in Qiu et al. (2013). Based on a test dataset of 25,796 posts on five ideological topics writted by 1,515 authors, the PSL achieves a F1 score of 74.0%, compared to a baseline SVM of 66.0%.

4.5 Conclusion

Over the course of the next three chapters, I present four models developed to analyze distinct aspects of online discussions and detect differing types of evidence of author stance. These are: a comment topic stance classifier (Chapter 5), an agreement classifier (Chapter 6), and a username classifier and an alignment prediction model for pairs of authors (both in Chapter 7). The outputs from these component models will feed into the integrated author topic stance classifier presented in detail in Chapter 8.

Chapter 5

Direct Stance Classification

5.1 Introduction

An obvious first step in the task of detecting the stance of a participant in an online discussion about a polarizing topic is to examine the individual comments posted by that author and look for evidence of expressions of his or her topic stance within the text of those comments. As discussed in Chapter 1, it is not the case that every post will give a direct indication of the author's underlying topic stance. Instead, a comment might be only tangentially-related to the main topic under discussion (or else be off-topic entirely), or it may be a response to a previous comment - possibly expressing agreement or opposition to another participant in the discussion - and thereby only offer an indirect signal to the author's topic stance.

However, in many posts there may indeed be a direct indication of the author's topic stance, which can be expressed in a host of ways. For instance, a post may include explicit first person assertions of support or opposition (e.g. '*I support the right of gay men and women to get married*') or statements about how the way of the world should be, according to the author (e.g. '*Everyone should be able to marry who they love*'). Commenters can also signal their topic stance by using sentiment expressions (e.g. '*Gay marriage is immoral and disgusting*'), or by choices they make in how they frame the topic, appealing to particular arguments to support their position, over others. For example, in a discussion on marriage equality, conservatives may make more references to morality or the bible, whereas liberals might be more likely to refer to the concepts of equality

and freedom. Fine-grained lexical choices can also indicate the stance of a commenter, when the author has chosen to use a particularly-loaded lexical item over a more neutral choice (e.g. '*partial-birth abortion*' as opposed to '*late term termination*'). This picture is often muddied, however, when a commenter makes use of concepts or language that is more typically associated with posters on the other side of the debate - either via direct quotation or paraphrase - in order to disagree, refute a claim, correct an assumption, and so on.

In this chapter, I describe a topic stance detection algorithm that classifies the stance (i.e. *pro* or *con*) of an individual comment with respect to a polarizing topic under discussion. I detect topic stance in a supervised machine learning setting, using binary classification. I train and test the performance of the system on a discussion on the topic of marriage equality, but I also describe how it could also apply to other comparably controversial topics. The system builds on previous work in the development of stance detection algorithms, and explores different types of feature representations of the comment to be classified. One such set of features aims to capture the underlying meaning of the text of a comment, via the extraction of the core propositional content of comment text. This methodology attempts to improve upon the limitations of a traditional bag-of-words approach to feature representation for text classification.

In the rest of this chapter, I begin by describing the related work in the area of stance detection. I then describe the external dataset used for training and testing the classifier, and how the features for the model were extracted. I show the results of experimenting with elements of the feature engineering and model design. I conclude with a discussion of the model errors, and make suggestions for future improvement.

5.2 Related Work

Classifying the stance of a given piece of text is a classic problem in natural language processing. The earliest work on stance detection focused on the debate setting of congressional floor debates (Thomas et al. (2006); Bansal et al. (2008)). In this work, supervised machine learning techniques were applied to try to automatically detect the political affiliation of a speaker, based on the text of the transcript of these debates.

The genre of online discourse differs greatly from congressional debates in terms of language use, however. Commenters tend to use colorful and emotional language to express their points, make use of sarcasm, throw insults, and question each other's assumptions and evidence. These characteristics make stance classification of online debates much more challenging. The first work looking at stance detection in this setting (Somasundaran and Wiebe (2010), Anand et al. (2011), Hasan and Ng (2013b)) explored the question of whether it is possible to automatically classify the stance of a single contribution to an online discussion forum with respect to a two-sided issue. This work uses features extracted from the textual content of the post (and possibly features extracted from the parent post), typically combining the traditional surface bag-of-words lexical features with hand-designed syntactic features or lexicons, but gives little if any consideration to the network structure generated by the interactions between users. Somasundaran and Wiebe (2010) trained a SVM classifier to predict the stances of individual posts using unigrams, arguing-based features (derived from a hand-built Arguing lexicon) and sentiment-based features (based on an external sentiment lexicon). They found that the simple unigram baseline was hard to beat, and the inclusion of arguing- and sentiment-based features together only marginally improved performance of the classifier. Their best performance for siding ideological debates ranged from 60.6% to 70.6% for four ideological topics (the second amendment, abortion, evolution, and gay rights) with an average of 63.9%, compared to an average accuracy of 62.5% based on unigrams alone. This range indicates that the methodology seems to generalize quite well across topics. Anand et al. (2011) looked at a different data set of ideological debate posts and explored a wider selection of feature types, including document statistics, post-initial cue words, counts of emotion and opinion words, syntactic dependencies, and the corresponding features from the comments parent. They concluded that the results showed, in general, that if the data were aggregated over all topics, the constructed features did not significantly beat the n-gram (unigram plus bigram) baseline. Their best results ranged from (depending on the topic) 54% to 69% accuracy. Hasan and Ng (2013b) experiment on the same dataset as Anand et al. (2011), and present slight improvements in accuracy using features based on automatically-extracted FrameNet semantic frames, and the application of an extra-linguistic constraint whereby all comments written by a single author must have

the same topic stance. They also indicate that classification accuracy can be improved by increasing the complexity of the model, either by taking a more fine-grained approach (that is, jointly modeling the stance of each sentence in a post as well as the stance of the overall post) or by jointly classifying sequences of post labels.

More recent work in stance detection takes on the related, although more complex, task of automatically recognizing arguments in online discussions, that is, the topic-specific reasons authors use to justify their stance on a particular issue. This is considered either as a stand-alone task, whereby a debate post is automatically classified as how strongly it is related to one of a pre-defined list of arguments (Boltuzic and Šnajder (2014)), to aid in debate summarization (Misra et al. (2015)) or as a subcomponent of the stance detection task (Hasan and Ng (2014)). In this latter work, the researchers demonstrate that based on a reason-annotated corpus of ideological debate posts from four domains, sophisticated joint models of stances and reasons - using essentially the same types of features used in their 2013 paper - can yield more accurate stance classification results than their simpler counterparts. I will not be considering automatic argument detection in this work, given the need for a corpus manually annotated for the presence of topic-specific arguments.

Finally, there was a raft of recent work related to the automatic detection of stance in Twitter data, as this was the sixth of the 2016 SemEval tasks (Mohammad et al. (2016a)). A training set of 4,870 English tweets was provided for stance towards six commonly-known targets in the United States ('Atheism', 'Climate Change is a Real Concern', 'Feminist Movement', 'Hillary Clinton', 'Legalization of Abortion', and 'Donald Trump'), and the task was to build a supervised stance detection classifier that would predict one of three classes (in favor of, against, neutral). The task received submissions from 19 teams, wherein the highest classification F-score obtained was 67.82. The best performing systems used standard text classification features such as those drawn from n-grams and sentiment lexicons. Some teams drew additional gains from noisy stance-labeled data created using distant supervision techniques. A large number of teams used word embeddings and some used deep neural networks such as RNNs and convolutional neural nets. However, none of these systems surpassed a baseline SVM classifier that uses word and character n-grams as features (Mohammad et al. (2016b)).

5.3 Data

The dataset used for the development of the stance detection model was compiled from two online debating websites: www.procon.org and www.debate.org. Both sites are online discussion forums covering ideological, social, political, and other topics. On both websites, for a given debate topic the page is designed to allow users to post any comments in favor of the issue on one side of the page, and comments against the issue on the other. In this way, commenters express their personal opinions on the matter literally on one side of the debate or the other, thereby explicitly declaring their *pro* or *con* stance. It is this action of self-labeling that makes this dataset ideal for training a text classifier, since no time-consuming or expensive human annotation is required to determine the topic stance of the comment.[1]

Both websites have wide coverage of the range of polarizing social topics that are considered in this dissertation, namely abortion, marijuana legalization, gay marriage, transgender rights, and so on. For the purpose of this chapter, I have scraped the comments data relating to the topic of marriage equality, since this is the topic of the www.politico.com development and test data sets on which the integrated author stance detection model is applied in Chapter 8. However, one could easily pull the comparable, self-labeled *pro* and *con* comment data on other ideological topics, and apply the same feature extraction and model training methods as explained later in this chapter.

I scraped a total of 1,150 comments written on the topic of marriage equality from the two debating websites, as of April 30, 2016 (ignoring any 'response' comments from www.debate.org). These were balanced evenly between those comments in support of and against gay marriage, with 575 comments of each type. The data were processed using the standard text normalization and NLP pre-processing steps outlined in Chapter 3. Comments ranged in length from a single sentence to fifteen sentences, with an average of 3.4 sentences. The average number of word tokens per comment was 57.2. I randomly selected 80% of the comments for training and validating the classifier, and held aside

[1]On the www.debate.org site, it is also possible for a user to post a comment in response to another comment. However, in practice this is not commonly done; the vast majority comments are posted directly on either side of the topic. This means that the language used in the posts is mostly directly relevant to the topic under discussion, rather than agreement/disagreement-type language that comes about though the interaction with other users, which would only add noise into the topic stance data.

20% of the comments for testing.

There is also a further source of data from the www.procon.org site that I use to develop features for the topic stance classifier. This data is a curated set of *pro* and *con* arguments that are typically used by debaters when discussing this topic. In the case of gay marriage, the website lists fifteen reasoned arguments in favor of marriage equality (including such reasons as *Same-sex couples should have access to the same benefits enjoyed by heterosexual married couples* and *Gay marriage is protected by the US Constitution's commitments to liberty and equality*), and fourteen arguments are commonly-used used by someone taking the opposite stance (such as *The institution of marriage has traditionally been defined as being between a man and a woman* and *Gay marriage is contrary to the word of God and is incompatible with the beliefs, sacred texts, and traditions of many religious groups*). Each point is followed by a short paragraph elaborating on the argument, giving more details and background supporting information. These arguments are written by the editors of the website, not user-generated, and are available for the same range of debate topics (abortion, marijuana legalization, etc).

I collected the *pro* and *con* arguments on the topic of marriage equality, and aggregated the texts of the arguments on each side into a single document. These data were processed using the standard text normalization and NLP pre-processing steps outlined in Chapter 3.

5.4 Model

In this section I describe the features used to train the topic stance classifier, and discuss the choice of learning algorithm. Since a number of the engineered features rely upon the *propositional content* of the comment text, I start by describing the methology by which such information was extracted.

5.4.1 Extraction of Propositional Content

A bag-of-words representation relying only on unigrams and bigrams cannot capture longer distance dependencies between elements in the text, and therefore would not reflect the propositional similarity between a sentence such as *'Being gay is not natural'*

and a related sentence with additional lexical material in adjunct positions: *'Being gay is really just not at all natural'*. The adverbs *really*, *just* and *at all* serve to emphasize the author's opinion, but essentially the two sentences have the same core propositional content. These intervening tokens will result in a very different bigram representation for the second sentence, obscuring the propositional similarity between them.

Another example of the shortcomings of n-grams relates to negation, particularly where the negation takes scope over an embedded clause. For example, consider the two following comments, differing only in the inclusion of a negation particle in the second: *'I do think that they should have the right to get married'* and *'I don't think that they should have the right to get married'*. The unigram and bigram representations for these two comments will be very similar, with a high degree of overlap, so this representation would not provide any features that a model could learn to be able to discriminate between the positive and negative example. An n-gram representation would not reflect the fact that the proposition expressed in the matrix clause of one sentence is the logical negation of that expressed in the other. What's more, the negation taking scope over the respective embedded clauses means that these embedded propositions are also opposite in stance, even though they are composed of identical lexical material. A more useful and powerful representation would be one that was able to capture the opposite polarity of the stances expressed in this pair of examples, rather than just the very high degree of similarity between their component n-grams.

To address this issue, I developed features that go beyond basic n-grams, and better reflect the underlying meaning expressed in the comment text, using syntactic dependency relations to capture non-local dependencies among the constituents of a sentence and thereby the underlying propositional content. In order to develop such features, I wrote code to extract the basic subject-predicate propositional content from a comment. Each comment can then be represented as the set of propositions that it contains.

For this purpose a proposition is defined as an 'SVO' triple, consisting of (i) subject, (ii) verb, and (iii) (optionally) complement. To be more specific, the third slot in the proposition triple is used to capture a direct object or other type of complement phrase immediately following the verb (and is left empty if none of these elements exist). The proposition also includes a polarity indicator, which equals -1 if the sentence contains a

negation marker, and +1 otherwise. Adjunct material such as adverbial and prepositional phrases are not included in the extracted propositions. In this way, the comment in (1a) would be represented as a set containing two propositions, as in (1b):

(1) a. I quite like gay people, but I don't support gay marriage.

b. {((I, like, gay people), +1), ((I, support, gay marriage), -1)}

My assumption is that by condensing a verbose comment into its set of core propositions, with negation explicitly indicated, it will be easier to compare and contrast the polarity of comments that contain broadly similar words.

Given the dependency parse of a comment provided by the pre-processing step described in Chapter 3, it is possible to identity the components of the proposition triples and the corresponding negation polarities. This methodology is described in the following section.

I make the fundamental assumption that there is a separate proposition associated with every verb in the comment. The first step, then, is to identify all the verbs in the text of the comment, based on the part-of-speech tags, as this determines how many propositions the comment contains. Any modal or aspectual auxiliaries (sentence elements with an *aux* relationship to the verb in the dependency parse) were identified, and prepended to the associated verbs.

For each verb, the subject was identified, being the word token in the sentence that holds an *nsubj* or *nsubjpass* relationship to the verb in the dependency parse tree. In a similar way, the associated object/complement slot was filled by searching through the sentence looking for a element that holds one of a subset of relations to the head verb in the dependency parse. This subset comprises *dobj* (direct object), *acomp* (adjective complement), *attr* (noun complement), *ccomp* (finite sentential complement), or *xcomp* (non-finite sentence complement). If there is no such element in the dependency parse (as in the case of a sentence containing an intransitive verb), the third slot remains empty.

For subjects, direct objects and noun complements, any preceding material in the corresponding noun phrase, such as a determiner or pre-modifying adjective (indicated by elements to the left of the head noun holding a *det* or *amod* dependency relation to it), were also extracted and prepended to the noun. This is done in order to distinguish between two sentences with the very same word token in the argument position but with a

very different meaning as a result of the modification (e.g. '*I support traditional marriage*' and '*I support same-sex marriage*'). Without including this pre-modifying material, both sentences would be represented by the same proposition, namely: (*i*, *support*, *marriage*). A similar approach was taken for prepositions, whereby the prepositional object (indicated by the element to the right of the head preposition holding a *pobj* dependency relation to it) was also extracted and attached to the preposition. For clausal complements, the third slot in the proposition was recorded as simply 'CC' or 'XC' (for finite and non-finite complements, respectively.)

The negation polarity value for a given proposition was set to -1 if there was a word (*no*, *not*, *never*, etc), holding a *neg* (negative) dependency relation to the verb, or if there was an element *no* holding a *det* (determiner) relation to either the subject, direct object, or noun complement of the verb. Special consideration was given to the negation polarity of propositions contained within embedded clauses. If the proposition extracted from the matrix clause had a first person pronoun subject, a cognition verb (such as *think*, *believe*) and itself had negative polarity, then this negation was passed through to the embedded proposition, as in (2):

(2) a. I don't think gay marriage should be legalized.

b. {((i, think, CC),-1), ((gay marriage, should be legalized, NULL),-1)}

In the case of sentences with conjoined direct objects, multiple propositions were generated, using the same subject and verb, and the conjuncts filling the respective object slots, as in (3).

(3) a. I respect gay men and lesbians.

b. {((i, respect, gay men),+1), ((i, respect, lesbians),+1)}

In the case of sentences containing subjects with conjoined verb phrases, the subject of the first proposition was carried over to the second, as in (4).

(4) a. The state should do its job and not interfere.

b. {((the state, should do, its job),+1), ((the state, should interfere, NULL),-1)}

I processed the set of 1,150 comments on the topic of marriage equality using the methodology described above to extract the propositional content. The average number of

propositions per comment was 9.4. I also processed the texts of the edited arguments for and against marriage equality in the same way. This generated a set of 343 propositions, 169 of which were from the set of *pro* arguments, and 176 from the *con* arguments. Only two propositions ('*marriage is institution*', '*same-sex couples to marry*') were represented in both the *pro* and *con* arguments.

5.4.2 Features

I experimented with a combination of feature sets, including the standard n-grams and other features that have been shown to be predictive in prior work on stance classification. I also developed novel features based on the propositional content of comments, as well as hand-crafted features that are described below. The features fall into four broad categories: (i) lexical features, (ii) proposition-related features, (iii) sentiment-related features, and (iv) other features. These are described in detail below.

5.4.2.1 Lexical features

(a) n-grams As has been shown repeatedly (a.o. Somasundaran and Wiebe (2010), Anand et al. (2011)), it is difficult to beat a stance classifier that just uses basic unigram and bigram features. Consequently, I extracted features for each training instance relating to the 2,000 most frequent unigrams and bigrams in the training set, allowing for the standard set of English stop words from the scikit-learn (Pedregosa et al. (2011)) feature extraction library.

(b) Modals The model included features that capture the usage of modal auxiliaries in a comment, as inspection of the data indicates that modals are often good indicators of stance (e.g. *'Gays cannot be allowed to get married'*, or '*Everyone should be free to marry who they choose*'). The two features reflect the total counts of the modals (*should*, *must*, *may*, *can*, and *could*) used in the comment, with and without negation.

(c) Ideological and political orientation indicators I developed a set of hand-crafted features that aim to detect the broader ideological (i.e. liberal or conservative) or political (i.e. Democratic or Republican) orientation of the comment's author, on the

basis that such characteristics are generally highly correlated with one's stance on the polarizing social issues that are the subject of this research. According to a recent survey by Pew Research Center (2017), the percentage of self-described liberals who support gay marriage is 85% in 2017 (slightly lower, at 79% as of 2015, the time the data in this dissertation were collected), compared to only 41% of self-described conservatives in 2017 (and 30% in 2015). With regard to political party identification, the percentage of self-reported Democrats in favor of marriage equality in 2015 was 66%, compared to only 32% of self-identified Republicans. So, although not every liberal or Democrat participating in these online forums will be in favor of same-sex marriage, and not every conservative/Republican will be against it, if we can detect with some certainty the ideological or political orientation of an author in the discussion, this may give a signal which can be leveraged in the topic stance classification task.

One way to gauge whether a comment is written by a liberal or a conservative is to look for language in the comment text that signals membership of, or opposition to, one of these categories. For example, a comment may include a statement of self-identification, such as '*As a proud liberal...*', '*I have been a card-carrying Republican since Reagan*', or '*We conservatives need to stick together*'. More likely, though, is that an author will express negative or disparaging opinions about members on the opposing side of the ideological fence, describing essential properties of the out-group, their habitual behaviors, and so on, for example '*Liberals are very unintelligent animals*' or '*Republicans don't lie - they "misspeak"*'.

Given the two ideologically opposed camps at play here, and given the relative sparsity of comments that contain these in-group/out-group mentions, the data suggest that all references to a left-leaning ideology should be collapsed under a single label, and the same approach taken for references to a right-leaning ideology. Consequently, I created two sets of orientation terms, reflecting these left- and right-leaning ideologies, respectively. The set of terms relating to the left comprises *liberal*, *democrat*, and *progressive*, as well as the diminutive forms *lib* and *dem*. Any word token in the comment text from this set was replaced by the token LEFT. The corresponding set of terms for the right consists of *conservative*, *republican*, and *GOP*, as well as the diminutive forms *con* and *repub*, and words from this set were replaced by the token RIGHT. I applied

the same methodology to the plural forms of these same terms, replacing them with the tokens LEFT-PL and RIGHT-PL, respectively. This latter move was motivated by the observation that negative statements directed towards those in the opposing camp generally use the plural rather than the singular form of the term. I then extracted the unigram and bigram features containing LEFT, RIGHT, LEFT-PL, and RIGHT-PL. The rationale for using bigrams was so as to be able to differentiate between inclusive statements using first-person appositives (e.g. '*We liberals are just more compassionate than you are*'), and ones containing second-person appositives (e.g. '*You conservatives are all the same*') or vocatives (e.g. '*Hey Republicans, how does this affect you?*').

(d) Punctuation and stylistic markers In line with other stance detection work in the genre of online debates, I include features that capture the counts of repeated punctuation tokens (??, !!, ?!), ellipses, and emoticons.

5.4.2.2 Proposition-related features

I developed a number of features for the model based on the core propositions extracted from the comment in the manner described above. The first step was to filter the set of propositions to discard any propositions that were contained within the context of a question in the comment text. This is because it is not obvious that propositions embeded in this irrealis sentential context could be taken to provide a reliable indication of the author's stance. For example, the comment '*Why do you think lesbians are unnatural?*' contains the propositional content ('*lesbians*','*be*','*unnatural*'). However, in the context of a question, a human reader can easily infer that the writer of this comment in fact does not believe that lesbians are unnatural, and is instead challenging an interlocutor who does hold this negative opinion. A similar rationale applies for discarding propositions found in the antecedent of a conditional clause (e.g. '*If gay marriage were legalized, then ...*'). Depending on the continuation of this sentence, this could indicate a positive or a negative stance on gay marriage. For the same reason, propositions that were extracted from inside a stretch of quoted text are also ignored.

Next, it is necessary to address the issue of dimensionality, since there are very many of the extracted propositions that appear just once or twice in the training set. In-

tuitively, it would be better to collapse semantically-similar propositions together, thereby reducing the number of unique proposition types and increasing the counts for the reduced set of SVO triples. I address this in a number of ways.

First, all inflectional forms of nouns and verbs were replaced by their lemmatized forms, thereby losing information reflecting number, tense, aspect, etc. Similarly, the aspectual auxiliaries 'be', 'do' and 'have' were dropped from the verbs (but the modal auxiliary verbs were retained).

Next, I developed a handcrafted list of topic-specific synonym sets. For example, for this topic, a commenter could use of any of the phrases 'same sex marriage', 'gay marriage', 'homosexual marriage', or 'marriage equality', to refer to marriage between two people of the same sex. Consequently, I create a synonym set including these terms, and replace any instance of these terms by the label SSM (for same sex marriage). Other synonym sets created for this topic were: MARRIAGE (capturing cases where the term 'marriage' had not already been subsumed into the SSM category, and including expressions such as 'the institution of marriage', 'matrimony', etc), HOMOSEXUALITY (including 'being gay'), HOMOSEXUAL (to describe gay individuals - either singular or plural - including 'homosexuals', 'gays', 'lesbian', 'the gay community', and various slurs), and HOMOCOUPLE (including 'gay couple', 'two men', 'two people of the same sex'). By synonymizing these frequently-used nouns and noun phrases, related propositions (as in (5a) and (5b)) are collapsed into the same underlying representation (as in (5c)), and reduces the number of unique propositions significantly.[2]

(5) a. ((gay marriage, be, right), +1)

b. ((marriage equality, be, right), +1)

c. ((SSM, be, right), +1)

Finally, any determiners and pre-modifying adjectives were dropped from the subject and object slots in the propositions, leaving only the head noun of these noun phrases. This reduced the number of unique propositions further. The semantic information encoded in the pre-modifying adjectives will not be lost - see the next section regarding sentiment-related features.

[2]Analagous hand-crafted synonym sets of terms would need to be constructed for a topic stance classifier built for a different polarizing topic, but this should not be onerous.

(a) Comment propositions To create feature vectors, I experimented with a 'bag-of-propositions' approach. First, each proposition triple was converted back to a text string by concatenating the subject, verb and object/complement, effectively creating a 'proposition trigram'. If the proposition had negative polarity, the prefix 'NEG-' was prepended to the verb. The proposition trigrams were then converted to feature vectors in the usual bag-of-words way. To compensate for the sparsity of these features, I also included corresponding proposition bigrams (subject+verb, verb+object), and proposition unigrams (subject, verb, object).

(b) Argument propositions I included additional features that reflect the level of propositional overlap between the comment text and the curated lists of arguments for and against marriage equality given by the editors of www.procon.org. These features count the number of propositions in the comment that are also in the propositional representation of the *pro* arguments, and the *con* arguments, respectively. It is expected that if the text of a comment aligns with the text of one of the arguments for or against marriage equality (at the propositional level of representation), then the stance of the comment will match the side of the argument.

5.4.2.3 Sentiment-related features

As described in Chapter 1, sentiment analysis is closely related to stance detection, in that the goal of both tasks is to classify the polarity of a given a piece of text. For stance detection, the predicted labels are *for* or *against* a particular position, whereas for sentiment analysis, the prediction is whether the the text conveys a *positive* or *negative* opinion, either in general, or towards a specific aspect mentioned in the text. Commenters in online debates often use sentiment-laden language to express their opinions on the topic (or a subtopic) under discussion. We have seen already many such examples in this thesis, such as '*I think that being gay is immoral and disgusting*'. The two strongly negative terms in this sentence give the reader a very clear sense of the author's stance on the topic of marriage equality. As described in the Related Work section above, sentiment-related features have been found in some cases to improve model performance. Consequently, I experimented with including different types of sentiment-related features in the model.

(a) Generalized sentiment I first developed a feature which represented the overall net sentiment of a comment, without regard to any specific target of that sentiment. This feature was designed to catch whether commenters on one side of the debate consistently differ with respect to their use of positive words and negative words than those on the other side. To do this, each sentence in the comment was passed through SentiStrength (Thelwall et al. (2012), refer to Chapter 3 for details) to obtain a positive and negative sentiment score for the sentence, and thus a net score. Sentence-level scores are aggregated so as to obtain a single net sentiment score for the comment.

(b) Topic-specific targeted sentiment I also developed features that attempted to capture the sentiment in the comment as it is directed to specific targets dictated by the topic under discussion. For this purpose, I used the same five concepts as were used in the development of the synonym sets for dimensionality reduction of the propositions described in the previous section, namely same-sex marriage, marriage in general, homosexuality, gay people, and gay couples. The intuition here is that a positive (or negative) sentiment expressed in a comment to one of these targets will likely be highly predictive of the stance of the comment on the issue of marriage equality.

I experimented with three types of topic-specific sentiment features. The first captures cases where a positive or negative sentiment is predicated of one of the terms in the target list defined above. For example, *homosexuality is unnatural* or *gay marriage is wonderful*. To determine this feature, I used the propositional representation of a comment and looked for cases where a term from the target list is in the subject slot of the proposition triple. For these cases, the verb and the object/complement are looked up in an external sentiment lexicon and the corresponding sentiment scores and added. For this purpose the MPQA Subjectivity Lexicon was used (Wilson et al. (2005), see Chapter 3 for details). The resulting sentiment score was negated if the proposition negation polarity is -1. If neither verb nor the object/complement appeared in the sentiment lexicon, the sentiment score was zero. If more than one proposition in a comment contains a target term as a subject, the returned feature was the sum of resulting sentiment scores.

The second topic-specific sentiment feature is reminiscent of the first and reflects cases where a term from the target list is preceded by a sentiment-bearing modifying adjective (e.g. '*I wish those immoral gays would just go away*'). To determine this feature

the dependency parse of the comment is examined to look for cases where an adjective appears in an *amod* dependency relation with a term from the target list. If so, the adjective is looked up in the MPQA sentiment lexicon, and the score is returned as the feature value. If more than one pre-modifying adjective in a comment is in the amod relationship with a target term, the returned feature is the sum of resulting sentiment scores.

The third topic-specific sentiment feature captures instances where the author of the comment expresses a sentiment-laden personal attitude towards a term in the target list (e.g. '*I hate the gays*'). To determine this feature, I looked for cases where a first person pronoun is in the subject slot of the proposition triple, and a term from the target list is in the object slot. The returned feature value is the sentiment score from the MPQA for the verb, and if necessary, negated and aggregated over multiple propositions in the same comment.

5.4.2.4 Other features

For good measure, I included features that reflect basic document statistics, such comment length (measured in sentences, words and characters), and average sentence and word length in the comment. These are generally included as feature sets in models for stance classification (Anand et al. (2011), Hasan and Ng (2013b)), although they generally have not shown to have much predictive power.

5.4.3 Model Design

For the choice of machine learning algorithm I experimented with Logistic Regression and Support Vector Machines, as both provide an associated probability (confidence) of a predicted class label. This will be needed for the downstream task of author stance detection, described in Chapter 8. The value of the model hyper-parameters (such as regularization type and parameter, and SVM kernel) was determined by five-fold cross validation on the training set. The results from the SVM were marginally (although not significantly) better than those from Logistic Regression, and so these are the results that are shown here.

5.5 Results

5.5.1 Significant Features

To investigate the differences in the language use in the two types of comments, I performed statistical tests comparing the feature values across both groups. For the binary features, I conducted chi-squared tests to compare the total counts of the feature for *pro* vs. *con* comments. For the features taking real values, the appropriate statistical test was instead an unpaired two-sample t-test on the mean feature values in the two groups. For each of the feature categories (lexical, propositional, sentiment, and others) I discuss below the features that differed most significantly between the *pro* and *con* comments.

5.5.1.1 Lexical features

With respect to the basic unigram and bigram features, the most significant terms relating to *pro* comments were: *is_love*, *deny*, *they_love*, *marry_who*, *people_should*, *love_is*, *to_marry*, *gay*, *religion*, *they_want*, *happy*, *straight*, *two_people*, *should_have*, *get_married*, *let_people*, *gender*, *freedom*, *just_because*, *love_each*, *regardless*, *homophobes*, *i_support*, *not_affect*, and *your_beliefs*. The corresponding list of significant terms relating to *con* comments were: *not_natural*, *and_women*, *one_man*, *sin*, *one_woman*, *nature*, *homosexuality*, *marriage_is*, *god*, *men*, *women*, *civil*, *adam_and*, *and_eve*, *against*, *union*, *bible*, *child*, *christian*, *female*, *homosexual*, *that_way*, *father*, *disgusting*, *god*, *'m_against*, *sick*, *hate*, and *institution*.

These two lists of significant terms give an insight into how the *pro* and *con* comments in the dataset differ in their framing of the debate on the matter of marriage equality. Comments in favor of marriage equality were generally used terms and phrases that evoked the concepts of love, equal rights, freedom, and fairness, whereas comments not in favor tended to talk more about traditional values, religion, family and the like. The two groups of comments also different in how frequently they used synonyms for the same concept, with the use of the term '*gay*' being correlated with *pro* comments and '*homosexual*' used more frequently in *con* comments. And as will be seen later, the *con* comments on average contained a greater proportion of negative sentiment terms.

As expected, there were some significant differences found between the frequencies of the engineered unigrams and bigrams containing terms reflecting the ideological left and right. The most frequent such ngrams that are characteristic of *pro* comments are: 'RIGHT-PL', '*you* RIGHT-PL', 'RIGHT-PL *think'* and 'LEFT', and those for *con* comments are: 'LEFT-PL', 'LEFT-PL *should*', and 'RIGHT'. These results are consistent with our intuitions that commenters often talk about, or address directly, authors in the other camp using these plural noun forms. Less frequently do commenters use language that mentions their own in-group, by using the singular noun or homonymous adjective form to self-identify or to refer to their value systems. This does appear to be a somewhat symmetrical pattern, with both *pro* and *con* comments showing similar levels of use of these ideological n-grams.

With respect to the other lexical features, I found that the use of modal auxiliaries (both with and without negative modification) are more characteristic of *pro* than *con* comments, but there were no significant differences in the stylistic features (counts of exclamation and question marks, ellipsis, use of emoticons, and so on) between the two types of comments.

5.5.1.2 Proposition-related features

There were some interesting significant differences between the *pro* and *con* comments with respect to the extracted propositions contained within them. There were a total of 257 unique proposition trigrams that occurred exclusively or primarily in the *pro* comments. The statistical tests revealed that the most significant of these propositions were: *love_be_love*, *i_support*_SSM, *people_have_right*, *they_want_xc*, *everyone_have_right*, SSM_*be_legal*, *i*_NEG-*see_reason*, *people_should-be_able*, *there*_NEG-*be_reason*, and *i_be_glad*. Correspondingly, there were 278 proposition trigrams that occurred solely or almost exclusively in *con* comments. The most significant of these were: *i_be_against*-SSM, MARRIAGE_*be_between*, SSM_*be_wrong*, SSM_NEG-*be_natural*, *i_have_nothing*, *god_make*_CC, *i*_NEG-*hate*_HOMOSEXUAL, *i*_NEG-*support*_SSM, and *i_be_christian*.

These propositions generally align with our intuitions about which side of the marriage equality debate a comment containing such a proposition would be on, ranging from the very explicit cases (e.g. *i_support*_SSM, SSM_*be_wrong*) to the more indirect ones

(e.g. *love_be_love*, *i_be_christian*). Even those propositions which do not contain topic-specific terms (e.g. *i_have_nothing*, *i_*NEG-*see_reason*) are readily interpretable when you consider the rhetorical devices that online contributors use to make their points more compelling. For example, an anti-marriage equality commenter could attempt to establish a general fairmindedness by prefacing her contribution with a concession, such as '*I have nothing against gays, but ...*'. On the other side, a proponent for same-sex marriage often index his intellectual acumen by appealing to reason and logic (e.g. '*I can't see any reason why...*'), thereby differentiating himself from a conservative who may hold his opinions based not on logical reasoning, but on tradition or scripture. This implies that such propositional features - when they are present - would be strong predictors of the topic stance of a comment. I explore this question in more detail in the next section when I discuss the results of the model at the prediction task.

For the sake of completeness, I mention that there were around 150 proposition triples that occurred roughly equally in *pro* and *con* comments. The bulk of these propositions did not contain much in the way of semantic content, either because of a pronomial subject (*it*, *that*, *you*), or a clausal complement (*cc*, *xc*). If we zoom in and look only at the propositional subjects, we can compare the *pro* and *con* comments to see what commenters on either decide of the debate prefer to talk about, as revealed by their choices of sentence topics. I found that the most frequent sentence subjects in *pro* comments were: *someone*, *you*, *people*, *love*, *everyone*, *two people*, *belief*, *homophobe*, *atheist*, *benefit*, *legalization*, *separation* and *constitution*. On the other hand, the most frequent sentence subjects in *con* comments were, in decreasing order: *child*, *it*, *i*, *god*, *man*, MARRIAGE, HOMOSEXUALITY, *bible*, *institution*, HOMOSEXUAL, *society*, HOMOCOUPLE, *family*, and SSM.

It is interesting to notice the different patterns of sentence subjects between the two classes. The anti-marriage equality commenters seem to prefer sentence topics that relate to the specific concepts falling under the marriage equality debate (that is, the concepts that were aggregated into synonym sets when the proposition features were created), or to religion and the family. They also tend to use a higher proportion of sentences with first-person pronoun subjects. On the other hand, the sentence topics used by commenters in favor of marriage equality are in some sense more abstract in that they include more

Table 5.1: Comment Stance classifier - Sentiment features

	PRO	CON	Signif
General Sentiment	-0.175	-1.086	***
Targeted Sentiment	0.094	-0.458	***
PreMod Adj Sentiment	0.0	-0.01	
First Person Sentiment	0.03	-0.036	***

generic terms (*people*, *love*), more quantified nouns (*everyone*, *someone*) or broader topic-related concepts, including legal or constitutional references. They also tended to use more sentences with second-person pronoun subjects. While this data set is too small from which to draw any strong conclusions, it would be interesting to see whether this same pattern - i.e. conservatives preferring a more focussed discussion, liberals appealing to broader concepts - is borne out in other discussions on other similarly-polarizing topics.

Lastly, consider the features which measure the overlap between propositions contained in the comments and the propositions extracted from the curated set of arguments in favor of and against marriage equality. Here, we see a clear, consistent pattern. There was a significantly higher proportion of *pro*-argument propositions contained in the *pro* comments (an average of 0.8 *pro*-argument propositions per comment), than in the *con* comments (0.1 *pro*-argument propositions per comment). The corresponding averages for the *con*-argument propositions are 0.2 (for the *pro* comments) and 1.1 (for the *con* comments). Both differences are highly significant, with $p < 0.01$.

5.5.1.3 Sentiment-related features

Table 5.1 shows the mean values of features for *pro* and *con* comments for the four sentiment-related features. I indicate three significance levels: * ($p < 0.05$), ** ($p < 0.01$), and *** ($p < 0.001$).

I found that there was indeed a significant difference between the sentiment scores for *pro* and *con* comments where the target of the sentiment is in the hand-crafted target list (gay marriage, gay people/couples, or homosexuality in general), or where the comment author uses a first person pronoun and expresses a sentiment-laden personal attitude towards a term in this same target list. In both cases, the *pro* comments show a net positive average sentiment score, and the *con* comments have a net negative score.

The feature relating to the sentiment of pre-modifying adjectives turned out not to be significant, given the very low frequency with which this feature fired.

Maybe surprisingly, I also found that the overall sentiment score of a comment, not even taking account of the target of the sentiment, was also correlated with the topic stance of the comment. On average, the overall average sentiment scores for *pro* and *con* comments were both negative. This likely reflects the genre of online debates, in which writers heatedly express their strong - and often negative - opinions about issues or entities under discussion. For this dataset, I found that the comments that are not in favor of marriage equality are significantly more negative than those in support of this issue. Whether the greater use of negative sentiment terms is characteristic of the language of conservative commenters more generally (and so whether we would see a similar pattern in debates on other topics) remains an empirical question.

5.5.1.4 Other features

Finally, with respect to the features relating to document statistics, I found as expected that there are no significant differences between the *pro* and *con* comments for these surface-level features. While *con* comments turned out to be slightly longer (60.4 words per comment, on average, compared to 53.9 words for *pro* comments), and this does rise to the level of statistical significance, given the size of the training set we should not read too much into this result as it is unlikely to be a meaningful difference that can be extended to other datasets or topics.

5.5.2 Prediction Task

This section evaluates the topic stance classifier with respect to the task of predicting the correct stance label on the unseen comments in the held-out test set. Given that the training set had an equal number of positive and negative instances, a trivial evaluation metric would be to compare to a coin-flip, with an expected accuracy of 50%. A more reasonable baseline is a simple n-gram (unigrams-plus-bigrams) bag-of-words model, in which the features relate only to the presence of lexical items in the training set, and do not include any more theoretically-motivated or engineered features.

Table 5.2 shows the results of the classifier using all of the features described

Table 5.2: Comment Stance classifier - Classification results

Features	Accuracy	P	R	F1	% Error Reduction
Random	50.0	50.0	50.0	50.0	
n-grams	68.7	69.7	66.1	67.9	
All	69.6	70.6	67.0	68.8	2.8
Best	70.4	72.0	67.0	69.4	5.6

above, as well as the best overall performance using a subset of the feature sets. Note that the n-gram features in the baseline model are not included in these models. The best model predicts the correct stance label in 70.4% of cases, with an associated F1-score of 69.4%. However, this is not a statistically significant improvement over the n-gram baseline. These results are consistent with previous work on stance detection discussed in Section 4.2, namely that classification accuracy generally falls in the range from 60% to 70%, and that, most of the time, sophisticated models trained on carefully engineered features most of the time perform no better than models using basic unigram and bigram features.

5.6 Discussion

The results above indicate that while there are some significant differences between *pro* and *con* comments with respect to carefully engineered, linguistically-motivated features, when it comes to the task of predicting the topic stance of a comment, the performance accuracy found by implementing a simple n-gram model is essentially just as good as a more sophisticated predictive model. I interpret this to mean that - for the topic of marriage equality, at least - the use of side-specific vocabulary is the major predictor of topic stance. In other words, people tend to present their arguments for or against the topic using certain 'frames'. On the *pro* side, as we have already seen, people like to talk about fairness, discrimination, equality and love, and the arguments put forth to support marriage equality often involve the psychological and economic benefits to individuals in society. On the *con* side, comments refer frequently to religion, tradition, and family, and the justification for being against marriage equality often invokes the 'slippery slope'

Table 5.3: Comment Stance classifier - Feature set ablation results

	Accuracy
Baseline - Majority	50.0
Baseline - n-grams	68.7
No modals	69.1
No ideological terms	68.2
No punctuation/stylistics	65.8
No comment proposition n-grams	56.6
No argument proposition overlap	65.1
No sentiment	64.8
No document statistics	69.9
All feature sets	69.6

argument - that it will lead to polygamy or incest, or that there are other alternatives available, such as civil unions. The consistent use of terms reflecting these frames appears to be equally predictive of comment topic stance as the more nuanced features aiming to capture the propositional content or the sentiment contained with a post. This is maybe a useful practical result, in that the model can be quickly trained and used to predict stances on unseen data without the need for a lengthy feature extraction process.

Even though the engineered features in the model did not in aggregate result in significant performance improvement over a simple n-gram baseline, nevertheless it is still illuminating to see the relative impact of the various feature sets. To do this, I ran the model with different combinations of the features to identify the contribution that each set of features makes to the overall performance. Table 5.3 shows the resulting model classification accuracy found by excluding each of the feature sets in turn.

The most significant observation from these feature ablation results is that the performance drops considerably when the comment proposition n-grams are excluded, from 69.6% down to 56.6% classification accuracy. This indicates that these proposition features are doing much of the same lifting as the lexical n-gram features used in the second baseline. This should not be surprising, given that the side-specific vocabulary used by the *pro* and *con* commenters that made the lexical n-grams to be so predictive of topic stance also appear in the propositional content n-gram features. On the one hand, the lexical features are able to discover significantly predictive unigrams and bigrams that

do not appear as core propositional content, such as adverbs and other adjunct phrases, whereas on the other, the propositional features are able to discover the significant longer distance relationships between words in a sentence, and therefore give rise to features that are more intuitive and interpretable. In this dataset, it looks as if the lexical n-grams have the slight edge over the propositional content features - the model performance using propositional features alone results in 67.1% accuracy, compared to 68.7% using lexical n-grams alone. It would be interesting to run this same analysis to compare the predictive power of these two types of feature sets on other datasets in this genre.

The second most significant set of features - as measured by the drop in model performance when the features are omitted - is the set of sentiment features. Once again, this conforms to the intuition that the expression of positive or negative sentiment towards a particular target is predictive of a commenter's stance. The reduction in accuracy tells us that the sentiment features are providing insights into the data over and above that which is given by the propositional content features, or any of the other feature types.

I carried out an analysis on a sample of 100 of the false positive and false negatives predicted by the model to get a sense of the types of errors that the model is making, and get an indication of how further features might be able to be developed. First, I found that nine of these errors were cases where the contribution appears to have been posted on the incorrect side of the debate, as in (6a) and (6b), or where the comment seems to exhibit both stances, as in (6c). This is a reminder that our methodological approach adopted to make use of this self-labeled data is not foolproof. Ideally, we would be able to get a much larger training set of self-labeled data than the data used to train the model in this chapter so that any errors arising from mislabeled cases are diluted.

(6) a. Love is love, no matter if it is a guy, or girl. [Actual=CON, Predicted=PRO]

b. I do not agree with being homosexual but I won't sit here and tell you how to live. [Actual=CON, Predicted=PRO]

c. Con if it forced on religious institutions that are against gay marriage. Pro if it has not effect on religion. [Actual=PRO, Predicted=CON]

Other than these mis-labeled cases, I identified a number of other consistent types of errors. First, there are the misclassifications as in (7) where a post contains language

that is generally typical of comments supportive of gay marriage, inclining the model to predict a *pro* stance, but the actual, seemingly-contradictory, stance is expressed in a contrastive clause later in the comment. Such errors suggest that it could be worth experimenting with features that rely on a discourse parse of the post.

(7) a. A gay person has the same rights a everybody else when it comes to marriage. But to pervert the institution that helps perpetuate a healthy society is wrong. [Actual=CON, Predicted=PRO]

b. I am all for LGBT rights and think they should have the right to be married if they so wish just like heterosexuals. However the supreme court‘s imposition of it is not something I approve of. I prefer free love to marriage anyway. [Actual=CON, Predicted=PRO]

Another category of misclassification errors relates to long comments in which the poster's stance is succinctly expressed in the very first sentence of the comment, but the remainder of the lexical material contains language that is more typically associated with the opposing viewpoint, such as '*oppose gay marriage*' in (8a) or '*Christian*', '*god*', and '*sin*' in (8b). This may suggest engineering features that take account of the position in the text in which n-grams or propositions occur, giving greater weight to those that are comment-initial.

(8) a. Gay marriage is wrong. Simple as that. The excuse that it has been going on for centuries is retired nonsense. Killing, war and disease has been going on for countless years also but we don't think it's right. I'm tired of many people using the same old statements and challenging topics on those who oppose gay marriage. From a religious and non-religious stand point. regardless of man, religion, or politics. I'm not on the matter of whether or not it should be legal. If it's legal for them to get married than let it be so. After all, religion is separated from the state. Parties do what they can to gain the best support from both the minority and the majority. However, just because it is legal does not make it right. Once again, this isn't an issue on should it be legal, or should it be okay, but an issue on acceptance and most of all moral standing. The reason many people oppose gay marriage is because they know it is wrong

or are still holding onto tradition. [Actual=CON, Predicted=PRO]

b. I am pro gay marriage because I am a Christian. People have a god given right to be able to choose to sin. They do not however, have a right to force other people to sin. So if gay marriage is legalized, there had better be protection for me to not participate at all in a gay marriage. We will let gays marry if they let us stay completely uninvolved. [Actual=PRO, Predicted=CON]

Yet another class of misclassification errors reflects the phenomenon of contributors deliberately using the language, lexical preferences or strategies of those on the other side of the ideological divide. This could include direct quotation, as in (9a) and (9b), or using 'us-versus-them' language, as with the repeated use of the pronoun '*their*' before the typically *pro* unigrams in (9b). Or, as in (9c), it could be citing the bible (which would typically be a rhetorical strategy used by an opponent of marriage equality), and then subverting the line of argumentation. This observation might suggest a more sophisticated treatment of quoted text and the detection of discourse coherence as part of the feature extraction process.

(9) a. I'm not 100% clear on the entirety of the issue, but I feel like this whole, "it doesn't matter who you love, as long as you love them" argument for gay marriage is really subjective. [Actual=CON, Predicted=PRO]

b. I hate this argument of "freedom" I keep hearing. Their choice, their rights, their liberty. It's wrong. the only reason this argument is even being held is because people have taught other people that you only need love. Love does not create marriage people. [Actual=CON, Predicted=PRO]

c. Leviticus said no homosexuality, but it also said no haircuts and not to wear two different types of fabric at the same time. So if we are going to follow Leviticus like a lost puppy, we'd better follow the rest of it. [Actual=PRO, Predicted=CON]

However, almost half of the misclassification errors were simply cases where there was not enough of a strong signal in the comment for the model to pick up on, even though human readers with their world knowledge and their ability to draw inferences from the propositions in the text. This is attributable in large part to the small size of

the training dataset as n-grams or propositional features which occur only once or twice in the corpus will not have any influence in the prediction task, even though they may be highly salient for human judges.

5.7 Conclusion

In this chapter I have described a system that classifies the topic stance (i.e. in favor or against a controversial topic) of comments posted in online discussion forums. I have shown that using features based on sentiment, the propositional content of a comment, and the use of ideological terms results in significant improvements compared to a majority baseline. However, they only perform about just as well as lexical features (i.e. n-grams) alone. I also highlighted statistically-significant differences between *pro* and *con* comments written on the topic of marriage equality, that - while not strong enough of a signal to influence the prediction task - nevertheless illuminate the different strategies that discourse participants adopt when expressing their opinion on this topic.

The comment topic stance classifier is one of the components that will be used in the overall task of author stance prediction explored in this thesis. To this end, the stance classifier was retrained on the entire set of www.procon.org and www.debate.org data. Because the topic stance model's performance has already been evaluated above, there is no further need to maintain a set of held-out test data. Instead, we can learn a more robust stance classifier using the larger set of combined training and test data, resulting in more reliable feature weights. This will optimize the performance of the downstream author stance prediction classifier, which is applied to the www.politico.com development and test datasets. I show the results of applying the comment topic stance classifier to the development and test datasets in Chapter 8, and discuss in detail how the predictions from this classifier contribute to and interact with the other components in the author stance detection task.

Chapter 6

Agreement Classification

6.1 Introduction

In any ongoing conversation other than the most rudimentary ones - be they spoken in face-to-face settings or written in online discussion forums - people express personal opinions about the matters under discussion, and inevitably other participants in the discourse will agree or disagree with these opinions. One view of such dialog is that the conversational record is part of the *common ground* of the discourse participants (Stalnaker (1978)). These discourse participants communicate through a set of dialog speech acts such as *assertions* and *proposals*, and *acceptances* and *rejections*. If an assertion proposed by one party is accepted by the other, the proposition becomes a mutual belief and it is entered into the common ground. On the other hand, if the assertion is rejected, the common ground is not updated. Discourse participants have many strategies at their disposal to carry out the dialog acts of acceptance and rejection. For example, other than by an explicit expression or by the negation of the assertion, the dialog act of rejection can be enacted indirectly by offering a logical contradiction, by denying or questioning a presupposition underlying the proposition, using implicatures, refusing to respond, and so on (Horn (1989), Walker (1996)).

More specifically in multi-party discourse, participants make social or 'alignment' moves with or against interlocutors to demonstrate their solidarity or to maintain their social distance (Bender et al. (2011)). Strategies for positive alignment include explicit expressions of agreement, praising, thanking, and positive reference to another partici-

pant's point, whereas negative alignment moves express disagreement with the opinions of another participant. This can include explicit expressions of disagreement, expressing doubt, giving sarcastic praise, or being critical, insulting, or dismissive.

The automatic detection of agreement or alignment between adjacent comments in online debates is a crucial element in the broader question of author stance detection taken up in this thesis. In a typical online threaded discussion, a good three-quarters of the posts in the discourse are written as responses to other comments, addressing points brought up in these prior posts, or directly addressing a previous comment author. In the dialogic context of discussions on controversial topics such as gun control, abortion and so on, posts often express their writers' ideological alignment with, or opposition to, a previous commenter's position. However, when these comment are decontextualized from the overall discourse these expression of alignment ('*I agree with you!*', '*You are so correct on this point.*') or opposition ('*Nope, you have it completely backwards.*', '*You're a total idiot.*') are impossible to interpret if we wish to infer the polarity of the author's topic stance on gun control (or whatever the topic of discussion happens to be). All we know for certain is that the author likely agrees (or disagrees) with the stance of the previous author. However, if we knew the previous author's stance, then we would be able to infer the stance of the current author. In this way, the automatic detection of agreement (or disagreement) between two comments offers us a hook by which we may be able to predict the ideological stance of an author, which is something we would not be able to do simply by looking at the text of that author's post without the context of the discourse in which it was written.

In addition to being a core component of author stance detection, agreement classification is a productive area of study in its own right. It can show insights into how conflicts arise between interlocutors, and strategies that discourse participants can use to resolve or escalate disagreements. The ability to detect agreement or disagreement between discourse participants has also been found to be useful for other tasks such as the detection of power hierarchies (Biran et al. (2012), Danescu-Niculescu-Mizil et al. (2012)), ideological subgroups (Abu-Jbara et al. (2012), Hassan et al. (2012)), and user interactions (Mukherjee and Liu (2013)).

In this chapter, I describe a model that identifies the level of agreement (or dis-

agreement) between two posts in an online discussion, where the second ('response', or 'reply') post was written as a direct reaction to the first ('parent') comment. I refer to this unit of analysis as a comment-response pair. I explore a rich collection of features extracted from the texts of the parent and response comments that may be able to distinguish between agreements and disagreements. These include lexical and stylistic features, discourse coherence, sentiment, and metadata such as comment length. I detect agreement in a supervised machine learning setting, using binary classification. The results indicate that it is possible to detect agreement in comment-response pairs (with disagreement being easier to detect than agreement). However, given the fact that human readers often need to draw upon their world knowledge and to rely on inferential processes in order to fully interpret a comment in the context, the agreement classifier does not come close to reaching perfect performance.

In the remainder of this chapter, I first discuss related work in the of area of agreement detection in online discussions. I then describe the Internet Argument Corpus, the annotated corpus used as a dataset for training and testing the agreement classifier. I explain the features utilized in the system, and show the results of experimenting with elements of the feature engineering and model design. The chapter concludes with a discussion analyzing the shortcomings of the model, based on an analysis of the model errors, and suggestions are made for future improvement.

6.2 Related Work

Some earlier work in this domain had focused on detecting agreement in spoken dialogs (a.o. Galley et al. (2004), Germesin and Wilson (2009)). However, that work is not directly comparable to the agreement detection considered in this chapter, for two main reasons. First, the data sets used were meeting corpus data (such as the ICSI meeting corpus and the AMI meeting corpus), and given the synchronous nature of the data it is not always possible to reliably identify dialog adjacency pairs among the multi-party conversations. Second, the prevalence of disagreement in these data sets was very low, which is maybe not surprising given the nature of the business meetings recorded in these corpora. Nevertheless, these early studies did motivate some the feature sets used in later

work on detecting agreement in online discussions, such as the development of lexicons of agreement and disagreement terms, turn-initial discourse markers, sentiment, and so on.

Of the more recent work considering the specific task of detecting agreement or disagreement in online discussions, the most pertinent are Abbott et al. (2011), Misra and Walker (2013), Mukherjee and Liu (2012), and Rosenthal and McKeown (2015). The first three works present supervised binary classification models, predicting the agreement between adjacent comments using various feature sets, including basic n-grams, sentiment polarity, and meta information such as comment length. Rosenthal and McKeown (2015) extends this basic approach to present a three-way classifier, including a neutral category of neither agreement nor disagreement. Abbott et al. (2011) use data from the Internet Argument Corpus (Walker et al. (2012b)) - the same corpus that is made use of in this chapter. They develop features relating to cue words, syntactic dependencies, and others derived from opinion and sentiment lexicons, and extract these from both the comment and its parent. On a balanced test set, the classifier achieves accuracies of up to 68% compared to a unigram baseline of 63%.

Misra and Walker (2013) sought to empirically test various theoretical accounts of the expression and inference of rejection in dialog, and study the effectiveness of topic-independent features, e.g. discourse cues indicating agreement or negative opinion regardless of the topic under discussion. Their model used sets of hand-crafted features motivated by theoretical predictions and based on the frequently occurring words and phrases in the training data that index denials, hedges, and so on. This work did not consider features extracted from the parent comment. Using a different section of the Internet Argument Corpus to that of Abbott et al. (2011), and a balanced test set, their results show that these theoretically-motivated features achieve 66% accuracy, compared to a unigram baseline of 60%. The most predictive features sets were found to be comment-initial discourse markers, as well as features based on punctuation.

Mukherjee and Liu (2012) expand on the modeling aspect of the problem and present a SVM combined with a generative topic model to automatically discover terms that are indicative of agreement or contention between adjacent posts. One of their main findings was that the rate of *accommodation* – the phenomenon where conversational participants adopt the characteristics of the other participants as conversation progresses

(Giles et al., 1991) – was generally more common in cases of agreement than disagreement. This includes the repetition of certain lexical material or syntactic structures, or semantic similarity between a parent comment and its response. Finally, Rosenthal and McKeown (2015) builds on the work of Abbott et al. (2011) and Misra and Walker (2013), using the dataset from the former (including the previously excluded neutral Q-R pairs), and some of the features from the latter, adding features representing aspects of the conversational structure (such as the lexical similarity between the comment and its parent), and accomodation devices, in which syntactic structures and lexical expressions from the parent are mirrored in the response comment. On the more difficult three-way classification task, and on an unbalanced test set, the classifier achieves an F-score of 54.4% (a macro-average over the three classes), compared to an n-gram baseline of 32.7%.

The approach to agreement detection in this chapter most closely relates to the work of Misra and Walker (2013), in that it is a binary classifier making use of a combination of hand-crafted and other engineered features. While a handful of new features are experimented with, and some are found to be predictive, the model presented in this chapter does not break any conceptual new ground. However, a decent agreement classifier is a prerequisite for the overall task of author stance detection presented in this dissertation.

6.3 Data

The dataset used for the development of the agreement classifier presented in this chapter was compiled from a subset of the publicly-available Internet Argument Corpus (IAC) (Walker et al. (2012b)), an annotated resource containing 390,000 online forum posts scraped from the debating website www.4forums.com on a range of topics such as abortion, evolution, and gun control. I use a section of the corpus comprising 10,001 quote-response (Q-R) pairs - that is to say, dialog excerpts in which a post to the forum quotes a passage of text from a prior comment in the discussion thread, and then provides a response to this quoted text. Each Q-R pair in the corpus was annotated by five to seven Mechanical Turk workers to indicate the level of agreement on a scale from -5 (total disagreement) to +5 (total agreement) between the response text and the quoted text

it was in reaction to, and the average agreement score for each Q-R pair was recorded. Examples of Q-R pairs at the two extremes of the agreement scoring scale are given as in (10):

(10) a. Q: Michael Moore tends to manipulate people, just in a different way than the President or the media does... not with fear, with knowledge and anger. R: Well said. I agree 100%. SCORE = +5.0

b. Q: Congratulations, Joe! I'm so happy for you. Sounds great. It's this aspect of our lives that the conservatives won't see... But anyway, more power to you, and I hope that you and your partner have many long and happy years together. R: Thank You! I think we will! SCORE = +4.67

c. Q: Hear my words I hate liberals and I will never change my mind about abortion. I try to love everyone but it's hard for me to love somebody who kills an innocent child. R: So when your God killed all of the helpless, little first borns in Egypt he too was another baby-killing liberal? You are very amusing and naive. SCORE = -4.67

d. Q: You should tell that to Gman. Most pro-lifers don't care about women, hence why they are pro-life. R: Really! You can prove that most pro-lifers don't care about women? ...it is idiotic thinking like this that makes me respect you less and less. SCORE = -5.0

As in prior work using this corpus, the scalar values were converted to categorical labels. Q-R pairs with an agreement score less than -2.0 were deemed to be examples of disagreement, and those with agreement scores greater or equal to +1.0 to be cases of agreement. The resulting distribution of agreement labels was 1,113 (11.1%) *Agree* and 3,452 (34.5%) *Disagree*. The thresholds of -2 and +1 were chosen to provide category frequencies that were not too skewed, and to reflect the observation that the Q-R pairs annotated with a small negative agreement score often did not exhibit unambiguous disagreement between the quoted text and the response. As can be seen there is a much higher number of disagreeing Q-R pairs in the corpus than pairs that agree. This is entirely consistent with the genre of online discussion forums where in general there is much more disagreement on display between discourse participants than there is agreement.

Table 6.1: Agreement classifier - Data sets

Label	Training Set	Test Set
Agree	890	223
Disagree	2,762	690
Total	3,652	913

The remaining 5,436 Q-R pairs, some 54% of the data, were cases of neither agreement nor disagreement. These neutral examples were omitted from subsequent modeling, and thus simplifying the problem to a binary classification task.

Of the resulting dataset, 80% of the Q-R pairs were randomly selected for training and validation, with the remaining 20% held out for testing. A summary of the training and test sets is shown in Table 6.1.

6.4 Model

In this section, I describe the features used to train the agreement classification model and discuss the choice of learning algorithm.

6.4.1 Features

I experimented with a combination of feature sets, including the standard n-grams and other features that have been shown to be predictive in prior work. I also developed novel features which aim to get at other ways in which online discussion forum participants can express their alignment with or opposition to previous contributions in the discourse. The different features are described below.

(a) n-grams I extract the standard bag-of-words (unigrams and bigrams) feature vectors for each Q-R pair in the corpus, ignoring stop words for unigrams. Further, in order to eliminate topic-specific expressions, I also ignore any n-grams occurring in discussions on fewer than three of the ten ideological topics (abortion, gun control, evolution, etc) that are represented in the corpus.

(b) Disagreement expressions In line with Misra and Walker (2013), I manually created a theoretically-motivated list of terms which at face value appear to be reliable cues to the identification of the speech act of rejection. This list of terms was compiled by introspection and an analysis of the most frequent unigrams, bigrams and trigrams in the training set. The list includes expressions such as '*disagree*', '*you're wrong*', '*oppose*', '*nope*', and so on. In total there are about 120 terms in this list. A feature is generated which counts the number of these disagreement terms that appear in the response post, with the proviso that the term is not adjacent to a negation marker (*no*, *not*, *never*) in the text. Furthermore, given that the dialog act of rejection is usually indicated at the beginning of the reply comment, I generate a second, related feature counting the number of disagreement expressions that occur in the first sentence of a post.

(c) Agreement expressions In a similar manner, I manually generated a lexicon of agreement terms that align with the speech act of acceptance, and this list includes expressions like '*agreed*', '*absolutely*', '*you're right*, '*correct*', etc, with a total of 45 terms. The two features (i.e. the count of the number of agreement expressions in the entire reply comment and in the first sentence, respectively) are further constrained to only count terms that are not followed by a discourse connective indicating a contrast (*but*, *however*, *yet*). This is to avoid counting cases where a post concedes a minor point in the comment it was responding to, but continues to disagree with the previous post.

(d) Discourse markers Previous work on discourse analysis (a.o. Fox Tree and Schrock, 1999; Groen et al., 2010, Galley et al., 2004; Louis et al., 2010) notes the particular pragmatic functions of different discourse cues, such as turn-initial *oh*, *well*, *really*, etc. Consequently, I created binary features that capture the initial unigrams, bigrams and trigrams in the response post.

(e) Discourse connectives The comment-initial n-grams described in (d) will also capture cases of the use of turn-initial discourse connectives, that can often signal the relationship of the comment with respect to its parent. For example, a response post that begins '*But...*' is highly likely to be presenting a point which is in opposition to a claim made in the prior post. In addition to the initial n-grams, I include five binary

features that indicate whether the first word of a comment is a discourse connective from the categories of CONTRAST (e.g. *but*, *however*), REASON (e.g. *because*, *since*), RESULT (e.g. *therefore*, *so*), CONTINUATION (e.g. *and*, *also*), and CONCESSION (e.g. *although*, *despite*). The full list of discourse connectives considered for each coherence relation type are taken from the Penn Discourse Treebank (Prasad et al. (2007)).

(f) Hedges Under Politeness Theory (Brown and Levinson, 1987), hedges are one strategy available for mitigating face-threatening acts, by softening a claim, lessening a criticism, or otherwise being more indirect in the use of language. It is possible, then, that the presence of hedges in a reply comment can be used as an indication of disagreement with the previous post. Consequently, I manually developed a lexicon of about 30 commonly-used hedging expressions, including '*maybe*', '*perhaps*', '*somewhat*', '*possibly*', 'I *wonder*', and so on. The related feature counts the number of hedge terms found in the reply comment.

(g) Insult expressions Given the adversial nature of many online debates, another common strategy that discourse participants use to indicate their opposition to a previous commenter is to use insulting language directed at them. To capture this way of expressing disagreement, I developed a lengthy list of insulting nouns and adjectives (e.g. *idiot*, *stupid*, *jerk*, *fool*, *asshole*, etc), that was seeded by introspection and then filled out by synonymous terms from the WordNet thesaurus. The related feature counts the number of insult terms found in the reply comment.

(h) Second person pronouns In a similar vein, I developed features counting the number of tokens of *you* and *your* in the comment. This feature is included in the model given the observation that comments overtly directed at the author of the preceding post are much more likely to express disagreement (or even hostility) than agreement. A related feature counts the number of second person pronouns in the first sentence of the reply comment, on the basis that such uses at the beginning of the comment are even more likely to be addressing the previous author, rather than a possible use of the generic *you*.

(i) Sentiment I aim to capture the overall sentiment expressed in a comment as follows. Each word in the comment tagged as a noun or adjective was categorized as strongly or weakly subjective, according to the MPQA subjectivity lexicon (Wilson et al., 2005), with the polarity flipped if the token occurred in the vicinity of a negation marker. Five sentiment features were recorded for each comment: (i) the strong positive polarity score (equal to the sum of the strongly subjective words of positive polarity), (ii) the positive polarity score (which also includes the scores of weakly subjective words of positive polarity), (iii) the strong negative polarity score (calculated in the same way as for the strong positive polarity score), (iv) the negative polarity score (ditto), and (v) a simple count of all negation markers (*no*, *not*, *n't*, *never*) in the reply comment. The five related features were also calculated for the parent comment. Binary features were then calculated to reflect whether the overall net sentiment for the comment (that is, the absolute value of the positive minus negative score) differed from or was the same as the overall net sentiment for the parent. A switch from a generally positive parent comment to a reply comment with a net negative sentiment might be a good indicator of a disagreement between the two comments.

(j) Interrogatives I included a feature which counts the number of sentences in a reply comment that end with a question mark. Inspection of the data indicates that a higher level of questioning speech acts within a comment is correlated with a commenter's disagreement with her interlocutor. Further, I included a binary feature which fires if the first sentence of a reply comment is an interrogative.

(k) Imperatives I included a binary feature which shows whether the first sentence of a reply comment is an imperative, on the basis that issuing a command to the previous comment author would generally be an indication of an oppositional stance. This feature was calculated by considering the part-of-speech tags of the post-initial trigram. The feature fires in the case of positive imperatives if the first part-of-speech tag is 'VB' (that is, the bare form of a verb) and in the case of negative imperatives if the third part-of-speech tag is 'VB', and it is preceded by *do* and a negation marker (*not*, or *n't*).

(l) Punctuation and stylistic markers The feature set includes counts of exclamation points, repeated punctuation (!!, ??, ?!), emoticons (postive and negative), use of all caps, expressive lengthening, and ellipsis, on the grounds that these may indicate heightened emotional language and therefore disagreement with the previous author.

(m) Length-based features This comprises a set of features based on length of post (number of words, number of sentences, average sentence length, etc). The theoretical motivation for including such features is that under the tenets of Politeness Theory (Brown and Levinson (1987)), disagreement is considered as a face-threatening act and thus a dispreferred response. Consequently, speakers expressing disagreement may have longer conversational turns, making use of more hedges, elaborations, and justification for disagreement.

(n) Features relating to the parent comment Other than the sentiment features, all of the feature sets described above consider only the text of the reply comment, and do not take any account of the text of the parent post. While it might be thought likely that the context in which a comment was written (that is, the text of the post that it responding to) would give helpful information in the ability to discern agreement or disagreement between the two posts, this turns out to be difficult to get at. Abbott et al. (2011) extract the very same features for both quote and response comments in the Q-R pairs in their dataset, and use these to train the model. However, they found that the excluding the Q-related features makes no difference to the model performance. Rosenthal and McKeown (2015) on the other hand, found that such accommodation features are mildly helpful in distinguishing cases of disagreement (although, interestingly, not for agreement or neutral cases). They also tried an alternative approach, taking account of the similarity of sentences (based on word overlap) between a reply post and its parent. They developed features that were based on unique words that occurred in similar sentences and whether just one of the similar sentences contained a negation marker. Their results found that such sentence similarity features were somewhat helpful, but the quantum of the boost to the performance of the classifier was not reported.

For this work I adopt a simpler heuristic approach, but one which is inspired by Rosenthal and McKeown (2015). For both the parent and the response comment

I extracted all the subject-predicate (S-P) token pairs, using the syntactic dependency parse of the comment as given by the NLP pre-processing. Specifically, for each main verb in the comment, the lemmatized form of the verb was recorded as the 'predicate', and the head word of the subject noun phrase that is in a *nsubj* dependency relation to the verb as the 'subject'. If the verb is the copula *be*, the predicate was replaced with the noun or adjective complement that follows it. The two resulting lists of S-P pairs were scanned, and just those cases where the same S-P pair is shared by the parent and response comment were retained. The polarity of each S-P pair was determined by looking for any syntactic negation that is syntactically attached either to the subject or the verb. I then developed two *propositional polarity* features for the model that count the number of shared S-P pairs that have the same polarity, and the opposite polarity, respectively between parent post and response comment.

6.4.2 Model Design

For the choice of machine learning algorithm I experimented with Logistic Regression and Support Vector Machines, as both models readily provide an associated probability of the predicted class label, which will prove necessary later on when the results of different classifiers are combined in the overall task of author stance detection. As discussed in Chapter 3, other binary classifiers, such as random forests, do not provide outputs that can be interpreted as easily. The choice of hyper-parameters (such as the type of regularization and the regularization parameter, and choice of SVM kernel) was determined by 10-fold cross validation on the training set. There were no significant differences in the cross-validated results between Logistic Regression and SVM, and so I present the results from the Logistic Regression model here. The model was trained using L1 regularization with a parameter of 1.0.

There is a mild imbalance in the class frequencies in the training set, with 76% of the cases being instances of disagrement, and only 24% agreement. To account for this, I experimented on the validation set with downsampling the majority class instances, generating new synthetic samples for the minority class, and adjusting the internal cost function weights, and adjusting the threshold to determine the final predicated class labels. However, none of these approproaches made a significant difference to the model that was

learned, and so ultimately I used an unbalanced training set. Rebalancing the test set was not considered, because this should reflect a more natural distribution of disagreement to agreement cases in comment-response pairs.

6.5 Results

6.5.1 Significant Features

To investigate the types of language used and different strategies for indicating agreement or disagreement comment-response pairs, I performed statistical tests comparing the feature values across both groups. For the binary features, I conducted chi-squared tests to compare the total counts of the feature for agreement or disagreement comment-response pairs. For the features taking real or negative values, the appropriate statistical test was instead an unpaired two-sample t-test on the mean feature values in the two groups.

A summary of the most significant ($p<0.05$) features from each feature set for the Agree and Disagree classes is shown in Table 6.2, in decreasing order of significance.

These results confirm that the features based on handcrafted lexicons for agreement expressions, disagreement expressions, and insult terms are each - not surprisingly - highly correlated with just one of two possible outcomes for a comment-response pair. With respect to the features based on the comment-initial unigrams, bigrams and trigrams, the analysis confirms that some discourse markers (*i know*, *it's just*) are more highly associated with agreeing comment-response pairs, whereas others (including *really*, *actually*, *so*, *you mean*) are more highly associated with discourse pairs that disagree. Other common discourse markers (*well*, *oh*, *i see*, and such) were not significantly different in their distribution between agreement and disagreement cases.

Some other results from the table to emphasize are that a small number of hedges are associated with disagreement turns, as predicted by Politeness Theory. However, the other prediction of this theory - that disagreeing responses would be longer than agreements - was not borne out, as none of the length-based features were found to be significant. With respect to the features that reflect turn-initial discourse connectives, those comments beginning with a connective from the categories of CONTRAST and REA-

Table 6.2: Agreement classifier - Most significant features

Feature Set	Class	Most significant features
Disagreement Expressions	Agree	-
	Disagree	*i don't think*, *i disagree*, *you're wrong*, *not correct*, *not right*
Agreement Expressions	Agree	*right*, *yeah*, *yes*, *correct*, *thanks*, *accept*, *agree*, *good*
	Disagree	-
Initial n-grams	Agree	*yes*, *i know*, *it's just*, *i think*
	Disagree	*really*, *no*, *actually*, *so*, *you mean*, *you know*
Discourse Connectives	Agree	CONTINUATION
	Disagree	CONTRAST, RESULT
Hedges	Agree	-
	Disagree	*somewhat*, *perhaps*, *maybe*, *i'm wondering*
Insult Expressions	Agree	-
	Disagree	*idiot*, *stupid*, *moron*, *fool*, *ass*, *ignorant*, *dumb*, *gullible*
Second Person Pronouns	Agree	-
	Disagree	*you*, *your*
Sentiment	Agree	-
	Disagree	PARENT_NEG_POLARITY, COMMENT_NEG_POLARITY
Interrogatives	Agree	-
	Disagree	COUNT_Q, S1_Q
Imperatives	Agree	-
	Disagree	NEG_IMPERATIVE, POS_IMPERATIVE
Punctuation/ Stylistic	Agree	POS_EMOTICON, !
	Disagree	?!, ??, NEG_EMOTICON, ALL CAPS
Length-based	Agree	-
	Disagree	-
Propositional Polarity	Agree	-
	Disagree	MISMATCH

SON are significantly more likely to be disagreements. This result for REASON coherence relations is maybe somewhat surprising. However, inspecting the data indicates that an available strategy commenters use to express disagreement with a prior comment is to take the line of argument started previously and to provide a continuation that asks for the reasoning behind the claim. Alternatively, a commenter may finish the thought expressed in the previous comment giving an absurd or sarcastic reason for why the previous events or claims had come about.

With respect to the sentiment features, the analysis showed that comments with net negative sentiment polarity are more likely to appear in disagreeing comment-response pairs, irrespective of whether it is the parent comment or the reply post that contains the more negative sentiment terms. Surprisingly, the mismatch between the net sentiment between the parent and reply comments was not a significant factor. The mismatch between the propositional polarity between parent and reply fired very rarely, but when it did, it was significantly more prevalent in disagreement comment-response pairs. However, the corresponding positive match between propositions in parent and reply comments was not found to be more highly associated with agreement cases.

Finally, some of the more prosaic significant findings were that second person pronouns are indeed used more often in disagreements than in agreements, as is the use of questions, imperative commands, repeated punctuation symbols, all caps, and negative emoticons.

It should be noted that while the features described above are the best ones at discriminating between the agreement and disagreement cases in the training data, in that there are statistically-significant differences between the two classes with respect to the counts or averages of these features, this does not necessarily mean that these same features will also be the most important in the prediction task of classifying the agreement status of comment-response pairs. It is very likely that - given the way that the model weights are learned during the machine learning training - some of these significantly different features will not have enough power in the model to make their influence felt over other, more powerful, features. This is particularly true for feature sets whose firing rate is low overall, such as the Imperative feature. In the following section, I point out which features are more important for the task of agreement classification.

Table 6.3: Agreement classifier - Classification results

		Agree			Disagree			Error
Features	Accuracy	P	R	F1	P	R	F1	Red (%)
Majority	75.6	-	0.0	-	75.6	100.0	86.1	
n-grams	87.4	70.3	83.9	76.5	94.4	88.6	91.4	
All	89.0	73.6	86.1	79.3	95.2	90.0	92.5	13.0
Best	90.3	76.2	87.4	81.4	95.7	91.2	93.4	22.6

6.5.2 Prediction Task

I now proceed to evaluate the agreement classifier as it is applied to unseen comment-response pairs in the held-out test set. In the results which follow, I show the performance of the classifier compared to two baselines, in one case using the majority class of 'Disagree' for all instances, and in the other based on basic unigram and bigram features. Since the test set is not balanced, the simple accuracy metric alone will not give entirely representative information about the model performance, and so I present the corresponding Precision, Recall, and F1 scores for both classes. I also show the percentage reduction in the number of errors for the experimental models over and above the n-gram baseline.

Table 6.3 shows the results of the classifier using all of the features described Section 6.4.1, as well as using the subset of the feature sets that provides the best overall performance results. The best model predicts the correct class category in 90.3% of cases, with an associated macro-average F1 score of 90.5%. Both the All Features model and the overall best model provide a statistically significant improvement (t-test, $p<0.05$) over the n-gram baseline, although there is no statistically significant difference between the two.

The experiments found that it is generally easier to correctly classify the disagreement cases in the test set than the cases of agreement. This conforms with the empirical observation that there are many ways to unambiguously express disagreement with or disapproval of a previous comment, whereas there are fewer ways to overtly express positive alignment, and instead agreement is often signaled more indirectly.

It is not possible to compare these results directly with the three most relevant studies (Abbott et al. (2011), Misra and Walker (2013), and Rosenthal and McKeown

(2015)). Misra and Walker (2013) trains the agreement classifier on a different data set, and further, use a balanced test set, with a correspondingly lower baseline model performance. Although Abbott et al. (2011) do train and test their model on the same underlying data as in this current work, they also use a balanced test set. Rosenthal and McKeown (2015) also work with the IAC Q-R pairs corpus, but they carry out multiclass classification, including a neutral category, and so the F1 scores they report are averaged of the three categories, and so cannot be compared to my results. However, the results are directionally the same, and appear to be broadly consistent.

6.6 Discussion

To test which feature sets in the model had the most impact on the overall prediction results, I carried out a feature ablation study. I found that the best performing feature sets in the prediction task were the disagreement and agreement expressions, the comment initial n-grams, discourse connectives, insult expressions, and the punctuation/stylistic features common to the genre of online discourse. This is broadly consistent with the findings of Abbott et al. (2011) and Misra and Walker (2013) (with the exception of insult expressions, which they did not consider). All of the other feature sets had either no effect on the model results, or else had a small negative effect on the classifier performance. Table 6.4 shows the resulting average of the F1 scores for the 'Agree' and 'Disagree' classes, with each feature set omitted.

It is not surprising that some of the features, such as interrogatives, imperatives, and the parent polarity, did not impact the model performance given the relatively low frequency with which they fire, as their effects were swamped by other features. More surprising were the length and sentiment features, the inclusion of which slightly dragged down the performance in the model.

Interestingly, the performance of the model using just the engineered features and omitting the n-gram features (average F1 score: 87.4%) was found to be only slightly less than the model using n-gram features alone (average F1 score: 87.8%). This result indicates that the set of engineered features described in Section 6.4.1 has almost as much predictive power in the classification task as the much larger, sparse feature set of n-grams.

Table 6.4: Agreement classifier - Feature set ablation results

Features Omitted	Average F1 score
No Disagreement Expressions	82.2
No Agreement Expressions	86.1
No Initial n-grams	85.5
No Discourse Connectives	87.0
No Hedges	89.0
No Insult Expressions	87.1
No Second Person Pronouns	88.4
No Sentiment	89.9
No Interrogatives	89.2
No Imperatives	89.3
No Punctuation/Stylistic	87.8
No Length-based	89.7
No Propositional Polarity	89.8
All Features	89.3
Best Model	90.5

This is maybe not an unexpected result, since the engineered features are theoretically motivated, interpretable, and applicable to other discussion threads regardless of topic.

While the results in the previous section show that the agreement classifier has relatively good performance, beating a standard n-grams baseline, there might still appear to be room for improvement. To explore this, I conducted an error analysis on a sample of the false positive and false negatives predicted by the best model. A few cases appeared to be genuine annotation errors in the corpus. Indeed, the level of inter-annotator agreement with respect to the judgments for agreement for the Q-R pairs among the IAC human annotators is not high, at 0.62 (Walker et al. (2012b)), reflecting the fact that detecting agreement can be inherently difficult.

Of the false negatives - true agreement cases that were incorrectly classified to be disagreement - I found a number of fairly obvious, straightforward explanations. For example, as the model is currently coded, a feature is triggered to fire if it sees one of a number of overt disagreement expressions in the first sentence of the comment. One of these disagreement expressions is the simple '*no*'. While this word is indeed often used to express disagreement with the previous comment, it clearly has many other functions in a sentence, and these other uses should mean that the unigram '*no*' is only a weak predictor

of disagreement. However, since the presence of this word is aggregated with the presence of other, unambiguous disagreement markers (such as '*disagree*' and '*wrong*') when the value of the feature is determined, this can bias the model to predict disagreement in instances where this is clearly not the case. Another error concerns questions. The motivation for including the presence of question marks in the comment as a feature was that it might be expected that interrogative statements within a comment would be directed towards a previous discussion participant, questioning his or her assumptions (or intelligence) and therefore signaling disagreement. However, the error analysis revealed that many questions inside comments are in fact rhetorical and are not oriented towards the previous commenter at all.

Within the false positives - true disagreement cases incorrectly classified to be agreement - I also found a number of unsurprising explanations. Many of these instances were predicted to be agreement based on the presence of '*yes*' at the beginning of the comment without a subsequent discourse connective indicating contrast. However, this simple feature gives entirely the wrong prediction if the semantic polarity of the comment it was responding to was itself negative. Consider the comment-response pair in (11), in which the comment-initial '*yes*' clearly refutes a claim made in the prior comment, and indicates disagreement, not agreement.

(11) a. Q: That is not even true!

b. R: Yes it is, police aren't obligated by the US Constitution to protect individual citizens. [SCORE = -4.83]

It requires a whole different set of machinery to detect the influence of logical (semantic) polarity as opposed to grammatical polarity in interpreting utterances as acceptance or rejection moves in dialog. Preliminary research in this area has been carried out by Schlöder and Fernández (2014), in which the authors develop a model inspired by work on the semantics of negation and polarity particles and test it on annotated data from two spoken dialog corpora. Their experiments indicate that heuristics which attempt to reflect the relative polarity of a proposal under discussion and of its response help to distinguish rejections from acceptances. However, their system cannot account for acceptance or rejection of a subclause, implicature rejections, rhetorical questions, and the like.

The other main cause of errors in this category was the use of sarcasm in replies. Responses such as "*Yeah right...*", "*Thanks for such a great explanation*", "*Good idea*", and the like, in the cynical world of internet commenting are more often than not intended sarcastically. However, the classifier trained on features based on the presence of the words *yeah*, *thanks*, and *good* is likely to predict these to be agreeing responses. The automatic detection of sarcasm in text is a complex and almost intractable issue, given that in many cases even humans are not able to reliably pick up on it. A principled way to tackle the issue of sarcasm would likely serve to increase the performance of the agreement classifier. However, it is beyond the scope of this dissertation.

6.7 Conclusion

In this chapter, I have described a system that detects agreement between comment-response pairs in online discussion forums. I showed that using features such as theoretically-motivated, handcrafted lists of expressions for agreement, disagreement, and insults, together with post-initial unigrams, bigrams and trigrams to capture discourse markers and coherence connectives, and punctuation and stylistic characteristics of the text, results in significant improvements compared to lexical features (i.e. n-grams) alone. I also highlighted statistically-significant differences between agreeing and disagreeing comment-response pairs, that - while not strong enough of a signal to influence the prediction task - nevertheless illuminate the different strategies that discourse participant adopt when seeking to perform the dialog acts of acceptable or rejection.

The agreement classifier is one of the sub-components that will be used in the overall task of the detection of author stances explored in this thesis. To this end, the agreement model was retrained on the combined set of the training and test IAC data, and the resulting model parameters were saved to disk. This means that the classifier with feature weights learned from this external dataset can be applied directly to unseen examples in the www.politico.com development and test datasets. I show the results of applying the agreement classifier to the development and test datasets in Chapter 8, and discuss in detail how the predictions from this classifier contribute to and interact with the other subcomponents in the author stance detection task.

Chapter 7

Other Indicators of Stance

7.1 Introduction

So far in this dissertation I have investigated the task of classifying the topic stance of a comment in an online discussion using features extracted from the comment text. In Chapter 5, such textual features were used as inputs to a classifier that predicted directly the stance of the comment. In Chapter 6, features were extracted from the text of both the comment and the previous comment in the discourse to which the comment in question was in response to. The features extracted from the comment-response pair were inputs to a classifier, resulting in a predicted agreement (or disagreement) between a comment and its parent. In this way, it is possible to indirectly infer the stance of a comment, given the stance of its parent, and the predicted agreement between the two comments.

In this chapter, I move beyond a fine-grained, bottom-up analysis based on the text of a single comment or comment-response pair, and investigate how top-down information - that is, features that are not extracted from the comment text - can be leveraged in the task of stance classification. I consider two main types of such top-down information. These are (i) metadata available from the commenters user profile, specifically, the choice of username; and (ii) indicators that reflect the tree-like structure of the threaded discussion, that is, who responds to whom and how often. I will show how both types of information can be used reliably to signal the ideological stance of commenters, and by extension, the stance of the comments that they contribute to the discussion.

7.2 Username Classification

7.2.1 Introduction

Commenting on most news websites and current affairs blogs is pseudonymous, meaning that a commenter must first create an account with a commenting platform (or first log in to a social network site, such as Facebook or Google+) before being able to post comments in online discussions. Typically, to create a user account, commenters must at a minimum choose a unique username or handle, provide a valid email address, and then submit other information to their user profile. This approach was introduced to counter the negative effects of purely anonymous commenting, which had been commonplace the very beginning of the social web when commenting was first introduced. These negative effects included the invidious practice of trolling (where anonymous commenters post highly offensive comments, including ad hominem attacks directed toward individuals or hate speech directed at entire classes of people) and spamming (where bots could automatically post advertising materials or other junk into a discussion forum). By requiring commenters to be registered with an account on the system, forum moderators could have a little more certainty that comments were being posted by humans, and so eliminate spam. Further, since comments could be traced back to a real email address, it was thought that this could alleviate to some degree (even if it could not eradicate completely) the issue of trolling.

The registration requirements vary across commenting platforms, but typically user profiles often include a display name (which can differ from the chosen username), an avatar (that is, a profile picture), and a short tagline in which they can enter a personal statement, self-description or other information.

The benefits of commenting with usernames for the task of stance detection are obvious, since multiple comments in one discussion can now be associated with the same author. This allows us to consider the more applicable task of classifying the stance of a comment contributor in a particular discussion, rather on the micro-task of classifying the stance of an individual comment taken in isolation, which probably has less directly useful application. There are a number of different ways that this information could be leveraged. One way might be to simply aggregate all of the comments written by a particular author

and running the composite comment through a stance classifier. Another way might be to classify all of the comments by one author separately and then apply some post-processsing of these results to arrive at an overall prediction for the author stance, for example, by taking the most frequently predicted stance among that author's comments.

A secondary benefit of users posting comments from accounts created on the commenting platform is that it gives rise to a community of users with persistent identities, who participate in discussions on different topics over a period of time. By applying a stance detection algorithm to a group of users across different discussions, it would be possible to see how the stance of individual users on a given topic may change over time. Furthermore, by noting which topics tend to have similar sub-groupings of users, it would be possible to learn the correlations between opinions on polarizing topics.

As noted in Lindholm (2013), "online nicknames convey intentionally-provided information based on how online users wish to present themselves in cyberspace." Consequently, a user's chosen handle may be a source of evidence that provides hints with respect to her general underlying social or political ideology. This in turn can be leveraged in the task of predicting the authors stance in any particular discussion on a given topic. This section takes up this latter question, one that to my knowledge has not been considered in the existing literature on stance classification.

By way of an example, a frequent commenter to articles on www.cnn.com with username *Conservative Patriot* has the personal tagline: 'Tea Party, Proud Republican, Pro-Liberty, Pro-Life'. From the username alone, our knowledge of the world gives us a strong sense of the stance that this person would take on divisive social issues such as marriage equality, gun control, or abortion rights. Indeed our judgment in respect of the latter is confirmed by the tagline, which literally spells out the commenter's political affiliation, as well as his stance on abortion. Intuitively, including this top-down information would result in a more accurate stance classification of the comments made by *Conservative Patriot* in a particular discussion than a model that only considered features extracted from the text of the comments themselves. The details of how such user profile information is folded in to the author stance detection problem are provided in Chapter 8.

For now, I concentrate on the specific task of identifying a user's ideological orien-

tation (that is, broadly speaking, whether they are on the right or the left of the ideological spectrum with respect to social and political issues) based upon their chosen username and the self-description given in the user profile tagline. To be sure, in a majority of cases I should suspect that a commenters username and profile tagline will not provide any evidence at all about an ideological orientation. Chosen usernames can reflect many different things, including users' real names, sports teams allegiances, characters in popular culture, or just random words or phrases. Moreover, since taglines are optional, the majority of registered users will not even have this information. So, I should not expect too many examples. However, the ones that are found are likely to be high precision cases.

There is not a great deal of research on the question of how participants in online discussion forums choose their user handles (or 'nicknames') to express their identity, or to index their affiliation to social groups. Bechar-Israeli (1995) analyzed 260 names on Internet Relay Chat and identified seven major types of usernames, namely: (i) people using their real name, (ii) usernames related to the self, (iii) nicknames related to the discussion platform, or to technology more broadly, (iv) names of flora, fauna, or objects, (v) plays on words and sounds, (vi) names related to figures in literature, films, fairytales, or of famous people, and (vii) names related to sex or other provocative topics. In total, the author found that some 45% of the nicknames were related to the self, meaning that in some way the nickname characterized the individual who used it, for example *shydude*, or *handsom*. Later, Stommel (2007) analyzed a small sample of usernames in an online forum for people with eating disorders, and found that their chosen handles play a important role in identity construction, and indexed personal characteristics such as smallness, weightlessness and negative self-evaluation, and in a few cases, self-confidence.

Most recently, Lindholm (2013) considers the choice and use of online nicknames as an example of the "efficient, cooperative use of language", in that they contribute to self-presentation, the negotiation of in-group identity, and "the co-construction of coherence." The author claims that online usernames can be viewed as being mini-propositions along the lines of "I am *nickname*" or "I like *nickname*", and applies Grice's maxims of conversation to the analysis of names taken from two online communities, around the topics of parenting and photography. The findings are that participants tend to adhere

most to the maxims of Quality - opting for the use of apparently genuine names and self-descriptions - and Relevance - providing information that they think is necessary to communicate about themselves.

Despite these analytical studies, there does not appear to be anything in the literature addressing the question of whether it is possible to automatically infer one's social, political, or ideological orientation from the analysis of a username as I present in this chapter.

The remainder of this section is organized as follows. First, I describe the characteristics of a corpus of usernames and user profile taglines collected in respect of individuals registered with the Disqus commenting platform. I then describe a methodology for inferring an online commenter's underlying ideological orientation based on the choice of username, highlighting and discussing the natural language processing challenges that this seemingly simple process presents. Finally, I evaluate the results of the model, comparing the predicted ideological orientation of users and usernames with the assessments of human annotators.

7.2.2 Data

I used the Disqus API to collect the publicly available information from the profiles of users who had participated in comment discussions on www.politico.com over the six month period from January 1 to June 30, 2015. For each user, I extracted the chosen username as well as any personal statement included in a tagline. Other information from the user profiles – such as the user's total number of posts, the forums participated in, the reputation, and information about other users being followed or following – was not collected, as it did not obviously lead to a path of identifying the ideological orientation. In all, user profile information from a total of 44,417 users was collected. Of these, around 10% (4,460 users) had included a tagline in their user profile, again potentially containing revealing information.

Before going further, it is necessary to take a detour to discuss various issues of orthography that make a data set composed of account usernames particularly challenging for standard natural language processing techniques. First, since usernames in the Disqus platform are constrained to be a string of alphanumeric characters with no spaces

or special characters, users make use of a number of strategies to create multi-word usernames. These strategies for marking word boundaries include the use of underscores or dashes (e.g. *long_john_silver*), or mixed (or 'camel') case (e.g. *LongJohnSilver*). Some users also use simple concatenation without spaces (e.g. *longjohnsilver*), or a combination of the above. While it is straightforward to use regular expression matching to detect the delimited examples or those using capitalization to indicate new words, it is much more challenging to automatically infer the spaces and thereby determine the appropriate tokenization of a username such as *longjohnsilver*. To do this, I developed an algorithm to find the most likely parse of the username string. First, I used a list of the most common 120,000 English words and their relative frequencies, from the Corpus of Comtemporary English (Davies, 2008). Under the Zipfian assumption that the frequency of any word is inversely proportional to its rank in the frequency table, I estimated the unigram probability for each word. Then, for each value of k from 1 to $(n-1)$, where n is the length of the input string, the algorithm generates all of the possible ways to insert k whitespace characters into the string, and then looks up the associated unigram probabilities of the resulting component substrings. If any of the substrings are not found in the corpus, this candidate parse is rejected. The algorithm outputs the parse that maximizes the product of the unigram probabilities for the candidate parses that it considers in the search space.[1] For example, the string '*themall*' will be parsed by the algorithm as ('*them*','*all*'), rather than the alternative ('*the*','*mall*'), since although the unigram probability of 'the' is much higher than the probability of 'them', the much lower relative frequency of 'mall' will drag down the probability of this parse below the probability of the alternative.

The other characteristic of this data that poses a particular challenge is the use of non-standard spelling, often for intended humorous effect or else to claim a name in the username space that has not already been taken. Common substitutions are 'R' for 'are', '4' for 'for', 'U' for 'you', '2' for 'two', and 'B' for 'be' (e.g. *conservativesRdumb*), zeros and ones for 'o' and 'i' (*t0x1c_avenger*), or more ad-hoc ones (*ca$h*, *wiggazz*, and *kkkrazyKansan*). I developed a set of replacement rules to standardize the orthography to the extent possible. After the preprocessing steps for tokenization and orthography

[1]The algorithm described above is computationally very expensive, so I optimized by using dynamic programming techniques in which the probabilities of the parses of prefix substrings are stored for re-use in subsequent passes through the iteration loop.

standardization, usernames were parsed into a list of word tokens (e.g. *longjohnsilver* → ['long','john','silver'], *conservativesRdumb* → ['conservatives', 'are', 'dumb']).

The corpus was then filtered to find cases that potentially signal the users underlying ideological orientation. To do this, I constructed a small, handcrafted list of orientation terms, reflecting vocabulary items that could have been chosen by users to reflect an underlying ideological orientation (that is, left- or right-leaning) with respect to social and political issues. The orientation terms relating to the ideological left were determined to be *left*, *liberal*, *progressive*, and *democrat(ic)*, along with their diminutive or abbreviated forms *lefty(ie)*, *lib*, *prog*, and *dem*, and associated plural forms. The corresponding terms relating to the ideological right were *right*, *conservative*, *republican*, *GOP*, and *tea* (as in *tea party*), along with the shortened forms *con* and *repub*.

The corpus was filtered to find all cases of usernames or personal taglines containing one of more of the orientation terms. Names containing orientation terms from both sets (e.g. *LeftCoastRightBrain*) were discarded. The resulting set was composed of 2,070 cases (around 5% of the total in the corpus). These comprised 1,090 cases (52.7%) containing a left orientation term and 980 cases (47.3%) containing a right orientation term. The set contained 308 cases (14.9%) that contained a plural orientation term, such as *dems* or *republicans* and 1,762 cases (85.1%) where the orientation term was a singular noun or adjective.

7.2.3 Rule-based Classifier

Given the corpus of usernames and taglines containing an orientation term (and thereby potentially indicating a user's underlying ideological orientation), the next step was to determine the polarity of this orientation. That is, would a person who selected this handle more likely be associated with someone on the left or on the right in the ideological spectrum?

This is not entirely straightforward. A username containing a token from the set of left orientation terms such as *liberal* may indeed indicate a left-leaning orientation (e.g. *ProudLiberalMom*) or the exact opposite (e.g. *filthy liberal scum*). Consequently, we need a way to determine whether the additional lexical material in the name supports or reverses the polarity of the orientation term included in the name.

To do this I developed a rule-based classifier that relies on an external sentiment lexicon to determine whether a positive or negative attitude is being expressed. For each name, I record the 'base polarity' of the orientation term (i.e. left or right) contained within the name, and then mask this orientation term. The resulting masked phrase is passed to a sentiment analyzer to determine an overall sentiment score. If the sentiment score is positive or zero, the base polarity is retained, and the model predicts this ideological orientation for this username. If the resulting sentiment score is negative, the original base polarity is reversed, and the model predicts that the username has an orientation opposite to that of the orientation term contained within the name. The same methodology is applied to the smaller set of taglines containing an orientation term. For the external sentiment lexicon, I used the SentiStrength resource developed by Thelwall et al. (2012), described in detail in Chapter 3.

It is illustrative to walk through a couple of examples, to make things more concrete. I show below the working of the algorithm for the two usernames *ProudLiberalMom* and *die_liberal_scum*. First, since both contain the orientation indicator 'liberal', they both have a predicted base orientation of 'left'. For *ProudLiberalMom*, the masked username is 'proud ____ mom', for which SentiStrength returns a net sentiment score of +2. Since the net score is positive, the predicted polarity is the same as the original base polarity. For *die_liberal_scum*, the corresponding net sentiment score is -3, resulting in a reversal of the base polarity, and a predicated orientation for this username of 'right'.

7.2.4 Results

The corpus of 2,070 usernames and personal taglines containing an orientation term was processed using the rule-based classifier described in the previous section. In total, 1,160 names (56.0%) were predicted to be users with a right-leaning orientation, and 910 (44.0%) were predicted to lean left. Encouragingly, the model made no inconsistent predictions for users based on the independent information contained in their usernames and taglines. A summary of the model predictions is shown in Table 7.1.

As may have been expected, the polarity of the orientation term in the name was highly correlated with the predicted orientation of the user. This reflects the tendency of users to choose a username that reflects their ideological orientation, rather than one

Table 7.1: Username classifier - Predictions

	Predictions		
Contains	LEFT	RIGHT	Total
LEFT term	831	262	1,093
RIGHT term	79	898	977
Total	910	1,160	2,070

which denigrates the other camp. In total, a full 1,729 out of 2,070 cases (83.5%) did not have the baseline orientation overturned by the algorithm as a result of the presence of negative sentiment terms. Drilling down further, only in 153 of these 1,729 cases (8.9%) did the SentiStrength analyzer return a net positive sentiment score. These are the cases discussed above, such as *ProudLiberalMom* and *LaughingLiberal*. Instead, in the vast majority (90.1%) of the cases that were not overturned, SentiStrength returned a net score of zero, indicating that there were no sentiment-laden words in the username. Such names reflected a wide range of strategies chosen by the users when selecting a username, such as concatenating the orientation term with a real name (*ConservativeJoe*), a geographical indicator (*SouthernDemocrat*), or other sentimentally neutral lexical material (*LeftLeaning*, *Conservative_Patriot*).

Of the 341 cases (16.5%) whose baseline polarities were reversed by the algorithm, reflecting the presence of negative sentiment words, a significantly greater proportion of these (76.8%) were instances of negative sentiment associated with left orientation terms (e.g. *NotFondOfLibs*, *Liberals_are_miserable*) and thus predicted to be chosen by conservatives, than the other way around, i.e. negative sentiment terms associated with right orientation terms (such as *Dump_the_GOP*, *republicansRevil*), predicted to be chosen by liberals.

To evaluate the results of the model, the predicted orientations for the usernames need to be compared with human judgments. I took a random sample of 250 usernames from the corpus and gathered responses from two human judges with respect to the orientation of the user who had selected that name. For each name, the judges were asked to say whether the likely orientation of the user was to the left (*liberal*, *democrat*, etc) or the right (*conservative*, *Republican*, etc) of the ideological spectrum, or whether it was not possible to tell.

The two annotators exhibited high levels of consistency on this three-way labeling task, agreeing on over 95% of cases, with a corresponding Cohen's kappa statistic of 0.78. There were only a total of 12 usernames for which the annotators disagreed with each other, and all of these were cases where one of the annotators was not able to tell with certainty the orientation, but the other annotator was. In other words, there were no cases when the two annotators had opposite judgments on a given name.

There were a further 23 usernames for which both judges agreed with each other that it was not possible to tell the ideological orientation of the author. There seem to be two main reasons for this. The first is that even though the username contains an orientation term which would normally indicate a clue as to an ideological orientation, these are in fact false positive hits. For example, in many cases, a name including the term 'right' is using this word not with its political connotation, but in other senses (e.g. *RightBackAtYa*, *Mom_is_always_right*). This also happens, although less often, for the orientation term 'left' (e.g. *LeftCoaster*).

The second reason that an orientation cannot be definitively discerned is that the username makes subtle use of wordplay, leaving an orientation implied but not unambiguous. For example, the username *RightBrainThinker* may be intended to simply invoke the popular psychological theory about the right hemisphere of the brain being more associated with creativity and the emotions, and thus imbuing the user with these characteristics. Or the choice of name could be more subtly implying that the user associates herself with the ideological right, and this identification forms the basis for how she views the world. Another example is *filthy_liberal*. The use of the singular noun 'liberal' would tend to indicate that the user is referring to himself rather than talking about someone on the other side of the ideological divide, but the highly negative adjective preceding it makes this conclusion less likely - unless of course the username were chosen ironically. It is much less clear how such cases could be detected and accounted for.

Putting aside these 35 disagreement and uncertain cases, the resulting annotated set consists of 215 usernames. Of these, 88 names were judged to be chosen by users on the left of the ideological spectrum, and 127 judged to be users on the right. These names were run through the rule-based classifier to obtain predictions of the stances associated with these users, which resulted in 96 predictions of liberal usernames, and 119 names

Table 7.2: Username classifier - Confusion matrix

	Predictions		
Actual	LEFT	RIGHT	Total
LEFT	74	14	88
RIGHT	22	105	127
Total	96	119	215

associated with a conservative worldview.

Table 7.2 shows how the model predictions for this set of usernames compares with the orientations of the users determined by the human judges. In total, the model makes the correct prediction in 179 out of 215 cases, which represents a classification accuracy of 83.3%.

7.2.5 Discussion

I analyzed the characteristics of the incorrect model predictions. Most of these cases seem to be where the SentiStrength sentiment analyzer does not have a sufficiently subtle touch, either because a loaded term is not in the underlying sentiment lexicon (e.g. *libs_are_commies*, *loons_to_the_left_of_me)*, or that a term is not deemed to carry any positive or negative sentiment given its normal sense, even though in context it is easy to infer a negative connotation (e.g. *Abort_democrats*, *Repubs_are_clowns*, *DishwaterTea*). There are even a couple of cases (e.g. *LiberalsMakeMeLaugh*) that returned a positive sentiment, thereby reinforcing the orientation of the baseline term. However world knowledge would lead most people who encountered this username to infer that its owner was not on the side of liberals.

It is not possible to account for such errors in a principled way, especially without extending or otherwise modifying the SentiStrength sentiment lexicon. However, since many of these model errors appear to involve the plural form of the orientation term, we can make a small easy change to the rules-based classifier to catch these cases. Previously the classifier retained the polarity of the base orientation term if the sentiment score given by SentiStrength was positive or zero, and reversed the polarity if the sentiment score was negative. The change now is that if the sentiment score is zero and the base

Table 7.3: Username classifier - Confidence of predictions

Orientation term in name	Confidence
'left', 'right'	0.844
Other singular term	0.885
Plural term	0.955

orientation term is plural, then the polarity is also reversed, as is the case when the returned sentiment score is negative. This reflects the empirical data, which shows that when a plural ideological term is used, it is usually in a disparaging phrase describing characteristics of the opposing side, rather than a reference to the user's own in-group. If this tweak is applied to the model, the number of errors reduces by over 30%, from 36 to 25, pushing the overall classification accuracy up to 88.4%.

Since the output of the username classifier is used as an input to the downstream integrated author stance model, it is necessary that the classifier outputs a level of confidence associated with the predicted orientation for each username. The rules-based classifier described in this chapter shows how a categorical label can be determined for each username, but does not naturally provide an associated probability that reflects the model's confidence in the prediction. Instead I will use an approximation based on the percentage of the model's correctly classified cases. However, rather than simply associating a confidence of 88.4% to all of the categorical predictions of the username classifier, we will achieve a greater degree of precision if the usernames containing singular and plural orientation terms are considered separately, as the model has differential classification accuracy across these categories, doing significantly better on the latter than on the former. For usernames containing plural orientation terms, the model has classification accuracy of 95.5%. For usernames containing singular or adjectival orientation terms, we can further analyze the model performance into that for usernames containing the basic orientation terms *left* and *right*, for which the model performed the worst (with a classification accuracy of only 84.4%), and cases where the username contained other singular orientation terms, such as *conservative* or *democrat*, for which the accuracy was 88.5%. Consequently, when handing off the predictions of the username classifier to the integrated author stance prediction model, I will use the confidence scores shown in Table 7.3.

Finally, I analyzed how the username classifier performed when applied to the community of discourse participants in the development and test marriage equality datasets. For the development set, there were 405 unique authors, and of these, 34 had a username containing one of the orientation terms. Of these, the username classifier decided that 17 were liberal names and 17 indicated a conservative orientation. This was the correct prediction in 30 out of 34 cases, a classification accuracy of 88.2%. For the 623 authors in test dataset, the username classifier found 51 usernames that contained an orientation term, and predicted 27 of these to have a left-leaning orientation, and 24 to be conservative. The model correctly predicted this orientation in 43 cases, giving a classification accuracy of 84.3%. These categorical predictions, along with the associated confidences, are passed along to the integrated author stance prediction model, described in Chapter 8.

7.3 Discourse Structure

7.3.1 Introduction

In this section, I investigate the potential indicators of author stance that are given by structural features of the discourse beyond the texts of the comment posts or the choices of partipants' usernames. To recap, online forums generally take the form of a *threaded* discussion, which means that one may choose to post a comment in response to any of the previous comments in the discourse. This results in a tree-like discourse structure for the overall discussion, with the comment (or different comments) that were posted directly at the top in direct response to the news article as the *root* comments, and any posts that are not replied to as the *leaves*. There are generally no technological constraints on the commenting platform with regard to the breadth or depth to which the discussion may grow. For instance, a particularly incendiary comment may receive tens or even hundreds of direct responses, resulting in a high branching factor and a broad tree-like structure. In addition, the discourse tree can become deep when a small group of commenters (or more often, just a pair of commenters) starts to engage in a back-and-forth conversation on a particular thread in the discussion.

It is this second characteristic of the structure of online discussions that provides

a potential foothold into the understanding of author stances. Intuitively, given the generally confrontational nature of online discourse, we may expect that when we see that two authors have engaged in a lengthy dyadic conversation within the broader context of a multi-party discourse, this pair of commenters are more likely to engaging in some kind of dispute or an exchange of differing opinions than they are to be persistently agreeing with or supporting each others' claims. The same intuition also applies in the case of any pair of commenters who have a lot of attested examples of interactions in the discourse - where one replies directly to the other - irrespective of whether these interactions occur along the single branch of a deep thread, or they take place in a widespread fashion across the breadth of the discussion.

The hypothesis investigated in this section is whether the probability of ideological agreement between two authors in a threaded discussion is inversely proportional to the number of attested interactions between the authors in the discourse. In other words, the more times we see Author A reply directly to a post by Author B, and vice versa, the higher is the likelihood that A and B are disagreeing on the topic under debate. In the analysis that follows, I make the assumption of the symmeticality of the relationship between A and B. In other words, there is no differentiation between the number of comments B leaves in response to A's posts, and the number that A leaves in response to B. The sum of these two counts is taken as the total number of interactions between this author pair, and use this number to predict the level of agreement between them.

In the remainder of the section I describe how the structural features of the www.politico.com development dataset were analyzed to build a regression model that predicts the level of author pair agreement in this dataset. I show that it is possible to build a model that closely mirrors the level of author pair alignment. Even though it may not be possible to infer the actual topic stance of each individual in the discourse using the results of this model, we will be able to get strong evidence as to whether two discourse participants are on the same or the opposite sides of the debate. (This is similar to the agreement classifier model presented in Chapter 6.) In Chapter 8, I will show how the regression model can be applied to the discourse structure of the test dataset, in order to get another set of clues that can help in the overall task of author stance prediction.

7.3.2 Data

The development dataset is described in Chapter 2. In summary, the discussion comprised 6,337 comments posted by 405 uniquely-identifiable authors. The human gold-standard annotation of these authors' stances showed that there were 204 authors with a *pro* (liberal) stance on the topic of marriage equality, 129 with a *con* (conservative) stance, and for 72 authors it was not possible to tell with any certainty. Among the set of 405 authors, there were 1,599 instances of attested of *author pairs* (a mere 2% of the 81,810 theoretically possible pairings of two authors from the 405 participants in the discussion). The two most interactive pairs in the development data were (*Dschwarpa*, *Dan*) and (*Dschwarpa*, *nogodintheconstitution*), who each had a total of 29 separate interactions across the entire discussion. On the other end of the spectrum, there were 855 author pairs who had just a single interaction in the discussion, with one of the pair replying to a post written by the other.

Excluding the posts written by the 72 authors for whom the actual stance could not be known, as well as the 12 other authors who did not have any interactions at all with other discourse participants, the resulting dataset comprises 5,572 comments posted by 321 authors (198 *pro* and 123 *con*), and 1,414 attested author pair interactions. Of these, 103 interactions (7.3%) were between two conservative commenters, 362 interactions (25.6%) were between two liberals, and 949 (67.1%) were between a conservative and a liberal. This can be compared with the probability that two people randomly drawn from a group containing 198 liberals and 123 conservatives would have the opposite stance, which is only 47.4%. This confirms the underlying assumption that in this genre of discourse, that when people interact, they are significantly more likely to disagree with each other than would be predicted by chance. As before, the number of interactions between any two authors ranged from one to 29.

A sample of the final dataset is shown in Table 7.4, showing the number of attested author pairs in the discussion for each number of interactions, the split of agreeing and disagreeing pairs, and the percentage of pairs which agree. We can notice that, in line with our intuitions, the probability of agreement rapidly reduces and approaches zero as the number of interactions increases. In addition, we see that any interaction at all between a pair of authors in the discussion decreases the likelihood of agreement between them

Table 7.4: Author-pair alignment model - Development set summary

Number of Interactions	Number of Author Pairs	Agreeing	Disagreeing	P(Agreeing)
1	739	311	428	0.421
2	284	88	196	0.310
3	119	23	96	0.193
4	70	12	58	0.171
5	41	7	34	0.171
6	37	5	32	0.135
...	...	...	...	...
24	2	0	2	0.0
25	1	0	1	0.0
26	2	0	2	0.0
29	2	0	2	0.0
	1,414	465	949	

below the chance fifty-fifty baseline. The data is visualized in the weighted scatterplot in Figure 7.3.2, where the size of the blob at each point is proportional to the number of attested author-pairs with that number of interactions between them.

7.3.3 Regression Model

I wish to predict the probability of agreement between two authors given the number of interactions between them. To do this, I fitted a regression model over the 1,414 data points summarized in Table 7.4. I experimented with both linear and exponential regression to find the curve of best fit through the data points. Unsurprisingly, the exponential model fit the data the best with an overall root mean square error (RMSE) of 0.047 (compared with 0.077 for the linear model). The final regression equation is shown in Figure 7.1:

$$p(Agree) = 0.53e^{-0.249n} \tag{7.1}$$

This model has two pleasing properties. First, for author pairs with no interactions in the discourse, it predicts an agreement level of around 0.5, which is consistent with the expected agreement rate of agreement for a pair picked at random from the whole

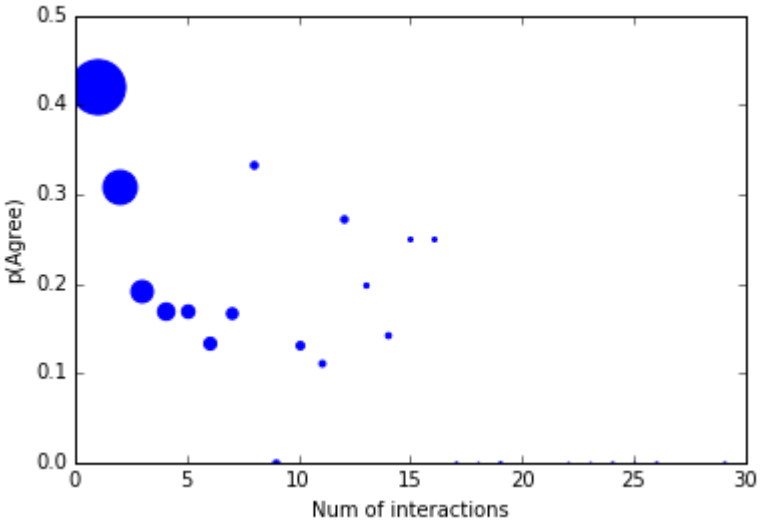

Figure 7.1: Author-pair alignment model - Development set agreement

set of authors. Second, the predicted probability of agreement approaches zero as the number of interactions between an author pair increases, and is negligible after 25 or more interactions. The resulting regression line is shown in Figure 7.3.3, superimposed on the weighted scatter plot.

I experimented with two additional features in the regression. The first looked at the elapsed time between when a comment was posted and when the reply came. The intuition here was that more instantaneous responses might signal additional 'heat' in a dyadic conversation that was indicative of more disagreement. However, I found that there was no significant difference in the average log transformed time between posts that agreed and comments that disagreed. I also explored whether changing the unit of analysis from the total number of interactions between a pair of authors to the lengths of the various *flurries* between them would result in a better model. By flurries, I mean the lengths of the distinct contiguous sequences of back-and-forth interactions between an author pair. However, I found that the RMSE for this model was greater than for that using just the total number of interactions.

It is possible to compare the predictions of the regression model to the actual levels of observed agreement between author pairs with a given number of interactions. To do this, the predicted probabilty of agreement is multiplied by the number of author pairs

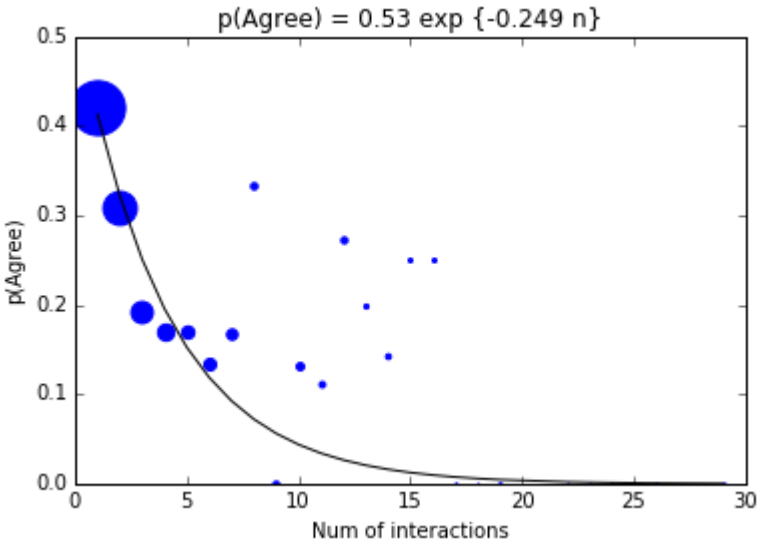

Figure 7.2: Author-pair alignment model - Development set predicted agreement

with that number of interactions in the discourse, to arrive at the predicted number of agreeing and disagreeing pairs, respectively. For example, if we look for the author pairs in the development dataset with exactly four interactions between them, we can see from Table 7.4 that there are 70 such pairs, 12 of which shared the same ideological orientation, and 58 of which disagreed of their topic stance. Substituting n=4 into the regression equation in (7.1) results in a predicted agreement probability of 0.20, and therefore an expected number of 14 agreeing author pairs from the set of 70. This can be compared to the actual observed number of 12 agreeing author pairs for this subset of author pairs. Running this same analysis across all of the possible numbers of interactions between the 1,414 attested author pairs results in a predicted number of 499 agreeing author pairs. This compares to the actual number of 465 agreeing author pairs in the dataset, meaning that only 34 of the instances would be misclassified by this model. This translates to an overall error rate of some 2.4%.

Finally, I apply the learned regression model to the www.politico.com test dataset. As described in Chapter 2, this discussion comprised 7,755 comments posted by 623 uniquely-identifiable authors, judged by human annotators to comprise 407 liberals, 127 conservatives, and 89 with an indeterminate stance. Excluding this latter group, as well as the other authors who did not have any interactions with other discourse participants,

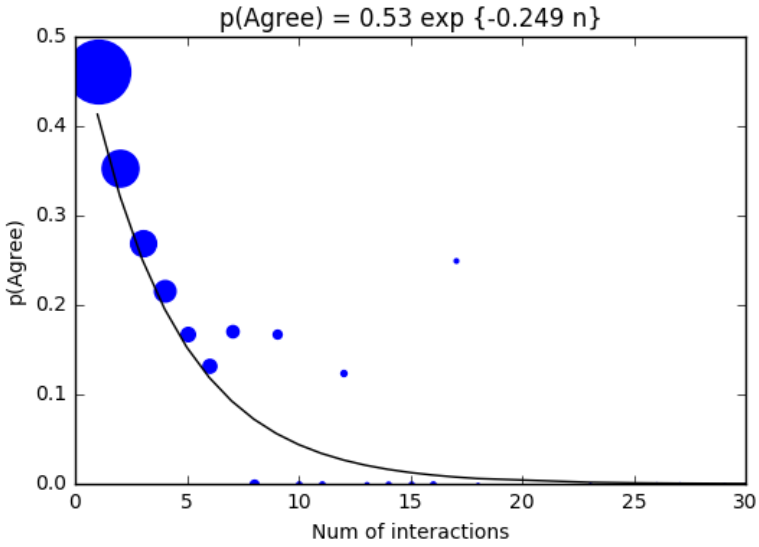

Figure 7.3: Author-pair alignment model - Test set predicted agreement

the resulting dataset comprised 497 authors (373 *pro* and 124 *con*), and 1,821 attested author pair interactions (669 of which were between like-minded authors, and 1,152 were across the ideological divide).

I then applied the same regression model that was trained on the development set to the test dataset, and the results are illustrated in Figure 7.3.3. The results show that the regression line using the parameters trained on the development set also results in a good fit on the test set data. The resulting RMSE is 0.045, which is almost exactly equal to the RMSE of the model trained and tested on the development set. With regard to the predicted numbers of agreeing and disagreeing author pairs, there is an overall error rate of 4.1%, meaning that some 1,747 of the 1,821 author pairs are correctly classified by the regression model. This error rate is a little higher than seen on the development dataset.

The results from this section show that there is clear evidence that the number of interactions in an online discussion between a pair of discourse participants is predictive of whether or not these two contributors ideologically agree or disagree. To be sure, the data set upon which the regression model was trained was fairly small, and a more robust model would benefit from more labeled data. However, getting the underlying author stances for the entire set of discourse participants is expensive, and I leave this task to future work. The predictions of the regression model are carried forward into Chapter 8,

where they are used as a component in the overall task of author stance predicton.

7.4 Conclusion

In this chapter I described a method to classify usernames as to the most likely political orientation (i.e. liberal or conservative) of the individuals who chose those names. This task involved some non-trivial text pre-processing to address non-standard orthography and word breaks that are prevalent in the choice of screen names. For usernames containing one or more from a set of orientation terms, the rule-based classifier is able to predict the correct orientation in 84% of cases.

The username classifier is one of the components whose outputs are integrated in the overall task of the detection of author stances. In Chapter 8, I apply the username classifier to the development and test datasets and show how the predictions from this classifier interact with the other sub-components in the author stance detection task.

This chapter also investigated the indicators of an author's stance that can be gained by looking at the tree-like structure of the discourse, without any regard to the actual comment text. I looked at the predicted level of disagreement between user pairs, given the number of interactions between them in the discussion. The predictions generated by considering the discourse structure form another of the subcomponents whose outputs are integrated in the overall task of the detection of author stances. In Chapter 8, I show how these predictions interact with the other subcomponents in the author stance detection task.

Chapter 8

Author Stance Prediction Model

8.1 Introduction

This chapter addresses the central question at the heart of this dissertation, which is whether it is possible to automatically detect the topic stance of authors participating in an online discussion with respect to a polarizing, contentious issue, such as being *for* gun control, *against* the legalization of marijuana, and so on. As we have seen, human readers often infer the ideological position of commenter without too much difficulty, using evidence available from a range of sources. These evidence sources include the textual content of the comments that the author has posted (such as their explicit expressions of opinion, or the choice of lexical items used or how their arguments are framed) as well as other, more indirect, indicators of stance that are available from looking at how a commenter interacts with other discourse participants (for example, expressions of agreement or disagreement, or the use of argumentative language). Yet more clues might be available from the discourse structure itself, such as the number or distribution of interactions between a commenter with other parties in the discussion, or other metadata.

In Chapter 4, I motivated the need for a collective classifier that takes into account these different potential sources of available evidence and jointly predicts author stances for all discourse participants simultaneously. In this way, evidence supporting an author's topic stance that comes, for example, from a vehement disagreement with a previous commenter can be used when there is no direct, topic-specific, evidence available from the text of the author's comment itself. Moreover, a collective classifier will consider the

entirety of the evidence available across the discussion at once, and will arrive at predicted stances for all authors that are the most consistent with the component classifier inputs - thereby smoothing out the noise from the predictions of those upstream models.

In the previous three chapters, I explored four sub-tasks that on their own could each be helpful in service of the overall author stance prediction task, targeting these different types of evidence that human readers are attuned to. These were the classification of the topic stance of an individual comment, the detection of agreement (or disagreement) between a comment and its response, the classification of an author's username, and the probability of stance alignment between pairs of authors. To recap, the corresponding models presented were:

Comment topic stance classifier (Chapter 5) This classifier predicts the stance (*pro* or *con*) towards the topic under debate for a comment posted in the discussion, based solely on the text of that comment. For every comment in the discouse, we have the predicted probability, p, of a *pro* comment topic stance. For an individual comment, a value of p greater than 0.5 indicates that the comment has a predicted *pro* stance, and a value less than 0.5 signals that it has a *con* stance.

Comment-pair agreement classifier (Chapter 6) The agreement classifier predicts the level of agreement between two adjacent comments in the discussion, based on the texts of the parent comment and the response post. For each comment-response pair in the discourse, we have the predicted probability, p, of the two comments agreeing with each other - that is, having the same topic stance. For a comment-response pair, a value of p greater than 0.5 indicates that a comment expresses agreement with the sentiment or stance expressed in the previous comment, either both *pro* or both *con* comments. A value of p less than 0.5 indicates disagreement, or that the two comments are taking opposite positions in the discussion.

Username classifier (Chapter 7) This rule-based classifier takes the username for a discussion participant and decides whether this chosen name reflects an underlying liberal or conservative worldview. In the majority of cases, this classifier will abstain from making a prediction. However, for those usernames containing a clue to the author's ideological

orientation, the classifier returns the probability, p, that this orientation is liberal, or left-of-center. Values of p greater than 0.5 indicate liberal usernames, and values less than 0.5 signal a conservative choice of username.

Author-pair alignment prediction model (Chapter 7) This model considers only the number of times a given pair of authors in the discussion interact with each other (that is, how often one author responds directly to a posting of the other), and then, without any information from the texts of their comments, outputs a predicted probability, p, that two authors share the same underlying topic stance. For every author-pair that interacted at least once, we have the probability of stance alignment. The value of p will be less than 0.5 for each author-pair attested in the discourse, and will decrease as the number of interactions increases, indicating a greater level of disagreement between pairs of authors that interact more.

In this chapter I bring everything together and present the details of the author stance prediction model. I describe a novel methodology that aggregates the predictions provided by the various component models, along with their associated confidences of those predictions, and based on this input, returns a prediction of the topic stance of every author in the discussion.

The remainder of the chapter is organized as follows. First, I present the formalism of the model. I then evaluate the performance of the model on two real datasets – the development and test sets described in Chapter 2. I conduct an extended analysis of the model as it is applied to the development set in order to investigate the relative contribution of each of the component classifiers, as well as to determine the value of a number of hyper-parameters in the model. For the independent test set, I focus on the model performance metrics (classification accuracy and such) and show how this varies for different subgroups of the discourse participants. I conclude with a discussion of some pleasing unexpected side effects of the model, including a potential way to identify so-called 'chaos creaters' in an online discussion, as well as the serendipitous discovery of a pair of duplicate usernames in the data.

8.2 Model

As discussed in Chapter 4, the intuition underlying the author stance model is that if we assume that all of the underlying component classifier predictions are correct, then we should infer the true stances of the authors that are most consistent with these inputs. In this way, it is analogous to maximum likelihood estimation, in that we want to end up with values of the underlying parameters (which in this case are the actual topic stances of the authors) that maximize the likelihood of the observed data (i.e. the predictions of the component classifiers).

In this section I present the details of the model formalism, with details of the representation of the data and component classifier predictions, the cost function adopted, and a description of the numerical optimization methodology used for model inference. I finish with an explanation of how the results are evaluated.

8.2.1 Data representation and model parameters

I formally define a discussion, G, comprising a total of n comments written by m different authors, as the tuple (A, C, U, V, W, P)[1], where:

- A is the set of m author IDs:
  $A = \{a_1, a_2, ..., a_m\}$

[1]To make this clear and concrete, I show here how the example discussion from Chapter 1 with eight comments written by four authors maps into the model formalism:

- $A = \{a_1, a_2, a_3, a_4\}$
- $C = \{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8\}$
- $U(a_1) = ConservativeKen$
  $U(a_2) = dj\_safari$
  $U(a_3) = davycrockett$
  $U(a_4) = happymofo$
- $V(c_1) =$ *It is my absolute second amendment right to own guns ...*
  $V(c_2) =$ *Wow ... paranoid much? ...*
  ...
  $V(c_8) =$ *Where are you getting that number from? ...*
- $W(c_1) = a_1$, $W(c_2) = a_2$, $W(c_3) = a_1$, ..., $W(c_8) = a_1$
- $P(c_1) = \emptyset$, $P(c_2) = c_1$, $P(c_3) = c_2$, ..., $P(c_8) = c_7$

- C is the set of n comment IDs:
  $C = \{c_1, c_2, ..., c_n\}$

- U is a function ($U : A \rightarrow text$) that maps author IDs to usernames
  $u_i = U(a_i)$ is the username of author a_i

- V is a function ($V : C \rightarrow text$) that maps comment IDs to comment texts
  $v_i = V(c_i)$ is the text of comment c_i

- W is a function ($W : C \rightarrow A$) that maps comment IDs to author IDs
  $w_i = W(c_i)$ is the author (or, writer) of comment c_i

- P is a function ($P : C \rightarrow C$) that maps comments IDs to other comment IDs
  $p_i = P(c_i)$ is the parent comment of comment c_i
  (NB: If c_i is a top-level comment, $P(c_i) = \emptyset$)

From these atomic elements, we can derive the following:[2]

- The set B of the q attested comment-response pairs in the discussion:
  $B = \{b_1, b_2, ..., b_q\}$
  $= \{(b_1^1, b_1^2), (b_2^1, b_2^2), ..., (b_q^1, b_q^2)\}$
  $= \{(c_i, c_j) : i, j = 1, ..., n \text{ and } P(c_i) = c_j\}$

- The set D of the r attested author pairs (or *dyads*) in the discussion:
  $D = \{d_1, d_2, ..., d_r\}$
  $= \{(d_1^1, d_1^2), (d_2^1, d_2^2), ..., (d_r^1, d_r^2)\}$
  $= \{(a_i, a_j) : i, j = 1, ..., m$, such that there is at least one $b \in B$ where the author of the first comment of b is a_i and the author of the second comment in b is a_j (or vice versa) $\}$

[2]For the example discussion, there are seven comment-response pairs (since c_1 was a 'root' comment, not written in response to another post), and four attested author-pairs:

- $B = \{(c_1, c_2), (c_2, c_3), (c_3, c_4), (c_1, c_5), (c_5, c_6), (c_5, c_7), (c_7, c_8)\}$
- $D = \{(a_1, a_2), (a_2, a_3), (a_2, a_3), (a_3, a_4)\}$
- $n_1 = 3$, $n_2 = 3$, $n_3 = 1$, $n_4 = 1$

- The total number of comments made by author a_i
  $n_i = \sum_j^n 1$ if $W(c_j) = a_i$

Now, consider the outputs of the four component classifiers. To make the mathematics simpler, I linearly rescaled the predicted probabilties from the component classifiers to a score in the range (-1,+1). For example, for the comment topic stance classifier predictions, predicted probabilities close to zero are transformed to a score close to -1, representing predicted *con* comments for which the classifier had very high confidence. Similarly, predicted probabilities closer to 1 become stance scores closer to +1, and these are the clear-cut cases of *pro* comments. Comments for which the topic stance classifier outputted a probability around 0.5 were equivalent to rescaled stance scores closer to zero. In a similar way, the predictions from the agreement classifier were linearly rescaled from probabilities in the range (0,1) to scores in the range (-1,+1), with more confident predictions of disagreements at the bottom end of the range, and high confidence predictions of agreement at the top end.

Now, define $\bar{X} = \{X_1, X_2, X_3, X_4\}$ as follows:[3]

- $X_1 : C \rightarrow (-1, +1)$
  $X_1(c_i)$ is the topic stance classifier prediction for comment c_i

- $X_2 : B \rightarrow (-1, +1)$
  $X_2(b_i)$ is the agreement classifier prediction for comment-response pair (b_i^1, b_i^2)

- $X_3 : A \rightarrow (-1, +1)$
  $X_3(a_i)$ is the username classifier prediction for author a_i

- $X_4 : D \rightarrow (-1, +1)$
  $X_4(d_i)$ is the author alignment prediction for the author pair (d_i^1, d_i^2)

Next, let's define a function S, mapping authors to their true stances. We assume that authors with a *pro* topic stance have a stance score of +1 and authors with a *con* topic

[3]For the example discussion, X_1 consists of eight predictions of comment topic stance, X_2 consists of seven predictions of comment-response pair agreement, X_3 consists of one prediction of username orientation, and X_4 consists of four predictions of author-pair alignment (in respect to the four attested author pairs in the discourse).

stance have a stance score of -1, consistent with the linearly rescaled component classifier predictions.[4]

- $S : A \rightarrow \{-1, +1\}$
  $s_i = S(a_i)$ is the true stance of author a_i

The set of stances, S, represents ground truth and serves as the yardstick against which the predictions of the model are evaluated.

8.2.2 Cost Function

With this formalism in place, we must now choose a cost function, $Z(S, \bar{X})$, that will be used to assess the degree to which the set of true author stances, S, is consistent with the set of component classifier predictions, $\bar{X}$. For a given configuration of S, this cost function should output a real value that provides a measure of how far off the predictions of the model are from the true author stances; assumed stances that are closer to the model predictions should incur a lower cost than those that are further away. Given than $\bar{X}$ represents the agglomeration of four component classifier inputs, the total cost is found by accumulating the costs associated with the different component predictions:

$$\begin{aligned} Z(S, \bar{X}) = &\text{ Cost with respect to } X_1 \\ &+ \text{Cost with respect to } X_2 \\ &+ \text{Cost with respect to } X_3 \\ &+ \text{Cost with respect to } X_4 \end{aligned} \tag{8.1}$$

The task then becomes a matter of finding the value of S that minimizes $Z(S, \bar{X})$, holding $\bar{X}$ constant.

After experimentation on the development set, I opted for a squared error loss function. This means that the cost incurred for each component classifier prediction is equal to the square of the difference between the predicted and the actual scores. In this way, misclassification errors (where the predicted and actual scores have opposite

[4]For the example discussion, S contains four elements, where S_1 is the true topic stance of *ConservativeKen*, S_2 is the true stance of *dj_safari*, and so on.

polarities) incur higher penalties than cases that are not misclassified. Moreover, more egregious misclassification errors for which the component classifier has higher confidence of prediction are penalized even more than cases that were closer calls.

For the comment topic stance classifier predictions X_1, the contribution to the overall cost function, Z, is calculated by iterating over all the comments in the discussion, and summing the squared difference between the predicted stance and actual stance of the author of the comment.

$$\text{Cost with respect to } X_1 = \sum_{i=1}^{n} [X_1(c_i) - S(w_i)]^2 \tag{8.2}$$

For the comment-pair agreement classifier predictions X_2, the situation is more complicated, given that the true agreement score for a given comment pair b_i depends on the stances of the two underlying comments b_i^1 and b_i^2 - and therefore on the stances of the authors of these two comments. By multiplying these two stance scores together we arrive at the desired result. This is where we see the benefit of rescaling predicted stance probabilities to scores that lie in the range from -1 to +1. If both underlying author stances are *pro* or both are *con*, the product of the associated scores will be positive, resulting in a correct positive agreement score between the two comments. On the other hand, if one of the author stances is *con* and the other is *pro*, the product of the associated stance scores will be negative, correctly reflecting disagreement between the two comments in b_i. The overall cost is calculated by iterating over all the comment pairs in the discussion, and summing the squared difference between the predicted and actual agreement scores.

$$\text{Cost with respect to } X_2 = \sum_{i=1}^{f} [X_2(b_i) - S(W(b_{i,1})) \cdot (W(b_{i,2}))]^2 \tag{8.3}$$

For the username classifier predictions X_3, the contribution to the cost function is calculated by iterating over the authors in the discussion for which the username classifier made a prediction, and summing the squared difference between the predicted stance and stance of the author. The penalty is upweighted by the number of comments that the author has left in the discussion.

$$\text{Cost with respect to } X_3 = \sum_{i=1}^{m} n_i [X_3(a_i) - S(a_i)]^2 \tag{8.4}$$

For the author pair alignment classifier predictions X_4, the calculation of the penalty is similar to that of the comment-pair agreement classifier in that it depends on the actual stances of the two relevant authors. The contribution to the cost function is then calculated by iterating over the known author dyads and summing the squared difference between the predicted and the actual scores for the author pair, where this latter element is found by multiplying the stance scores for the two authors.

$$\text{Cost with respect to } X_4 = \sum_{i=1}^{g} [X_4(d_i) - S(d_{i,1}) \cdot S(d_{i,2})]^2 \tag{8.5}$$

Putting this all together results in a total cost function, Z, as shown in 8.6:

$$\begin{aligned} Z(S, \bar{X}) = & \sum_{i=1}^{n} [X_1(c_i) - S(w_i)]^2 \\ & + \sum_{i=1}^{f} [X_2(b_i) - S(W(b_{i,1})) \cdot S(W(b_{i,2}))]^2 \\ & + \sum_{i=1}^{m} n_i [X_3(a_i) - S(a_i)]^2 \\ & + \sum_{i=1}^{g} [X_4(d_i) - S(d_{i,1}) \cdot S(d_{i,2})]^2 \end{aligned} \tag{8.6}$$

All that remains is to find the value of S which minimizes $Z(S, \bar{X})$.

8.2.3 Optimization

We could take a brute force approach to find the value of the vector S that minimizes the cost function, by calculating the total cost associated with every possible combination of author stances, and choose the configuration that minimizes this value. However, this is not tractable for a realistic dataset, as there will be an exponentially large number of combinations to take into account (=2^m possible outcomes, where m is the number of authors in the discussion). This is not computationally feasible.

To get around this, I relaxed the assumption that each author's stance, s_i, must take a binary value (-1 or +1), and instead assume that the stance can take all real values lying between -1 and +1. This makes the cost optimization problem easier, since we can

now apply standard numerical optimization techniques. This assumption has a nice side effect too, since it allows for finer gradations in the predicted stance than a simple black-or-white decision, aligning with the intuition that while two authors may have the same overall stance on a given topic, one may be only moderately conservative, say, whereas the other has a more extreme right-wing point of view. Another advantage of the real-valued predicted stance is that is can be used to provide a proxy for the confidence of the predictions of the model.

The numerical optimatization method used in this work to find the best vector of author stances, S, is a variant of the Broyden-Fletcher-Goldfarb-Shanno ('BFGS') algorithm. BFGS is an iterative hill-climbing algorithm for solving non-linear optimization problems, and since being first proposed by Malouf (2002), has widely used for parameter estimation in machine learning. The algorithm seeks to find the maximum or minimum value of an objective function, and it does this by traversing through the search space and, at each iteration, moving along the path with the steepest gradient. The full BFGS algorithm presupposes a convex, twice-differentiable, objective function, which is not always the case. Instead, the *limited-memory* variant of BFGS (Liu and Nocedal, 1989) is a more computationally-efficient implementation which calculates gradients only for representative sample vectors. L-BFGS has been shown to give good results even when the underlying function is not differentiable. The *bounded* variant of the BFGS algorithm (Byrd et al., 1995) imposes additional linear constraints on the variables to be solved for, in this case the fact that the stances must lie between -1 and +1, rather than being any real number. For this task, I used the L-BFGS-B algorithm as implemented in the SciPy Python library.

8.2.4 Evaluation Metrics

The model's predictions of author stances, S, are then evaluated by comparing them with the known stances of the discussion participants, as was determined by human annotators (described in Chapter 2). However, since these ground truth labels are categorical (i.e. *pro*, *con*, or *can't tell*), we first have to re-binarize the model predictions, according to the sign. Thus the model predicts that author a_i has a *con* stance if the predicted stance s_i is negative, and a *pro* stance if s_i is positive. We end up with a 3x2

contingency table, showing the ground truth stances against the stances predicted by the model. From this we can derive the usual classification evaluation results, such as accuracy, precision and recall, and F1 score, as defined in Chapter 3.

Another way to evaluate the model results is to measure the cross entropy error, which uses the predicted model probabilties, not the predicted class labels, when comparing against the true author stances. The intuition here is that we would want to favor a model that had stronger confidence associated with its correct predictions over another a model that made the same number of correct categorical predictions, but with weaker confidence. The formula for calculating the cross entropy error is given in 8.7 (where s_i and $\hat{s_i}$ are the true and predicted stances of author i, respectively).[5]

$$\text{Cross Entropy error} = -\frac{1}{m}\sum_{i=1}^{m}[s_i \cdot log(\hat{s_i})] \tag{8.7}$$

While the numerical value of the cross entropy error for a given model is not directly interpretable in its own right, it does allow for a more nuanced comparison of two competing models than simple classification accuracy. For instance, a model which predicts the correct author stances in all cases with high confidence will have a lower cross entropy error - and so be a better fitting model - than another whose predictions were much less confident, with predicted probabilities that are all close to 0.5 but just happen to be correct side of the decision boundary for every instance.

8.3 Results

I now show the results of the model as it is applied to the development and test data sets described in Chapter 2. I first evaluate the model's performance on the development set, and then use this dataset to explore the interactions and relative contribution of each of the component classifiers. I also use the development set to tune a couple of the model's hyper-parameters. For the test set, I look at the model performance metrics to confirm that the model is generalizable to a different discussion (on the same topic).

[5]For the calculation of cross entropy error, the stances must be linearly rescaled back to lie in the range from 0 to 1.

Table 8.1: Component classifiers - Predictions for development set

Component Classifier	Number of predictions	Authors covered
Comment Topic Stance (X_1)	6,337	405
Comment Pair Agreement (X_2)	4,839	381
Author Username (X_3)	34	34
Author Pair Agreement (X_4)	1,598	381
All predictions	12,808	405

8.3.1 Development Set results

As a reminder to the reader, the development dataset set relates to a discussion on the topic of marriage equality on the website www.politico.com on June 26, 2015. The online discussion comprised a total of 6,337 comments made by 405 unique authors (excluding a small number of comments that were left by authors with the username *Guest*). It is the *pro* or *con* stance on the issue of marriage equality that is the goal of the model. As described in Section 2.2.2, I collected human judgments of the stances of these 405 authors, and the results were 204 authors were judged to be in favor, 129 against, and for the remaining 72 authors it was not possible to judges to determine their stance with certainty.

The entirety of the discussion was passed through the four component classifiers and the resulting predictions were taken as the inputs to the integrated author stance prediction model. The number of each type of these component classifier predictions is as shown in Table 8.1.

The low number of authors included for the Username classifier reflects the small percentage of usernames in the dataset that contain an indicator of their political stance. The number of authors included for the Author Pair Agreement classifier reflects the subset of the total authors who have had an interactions with at least one other discussion participant.

Given the inputs described in Table 8.1, the model predicted a total of 231 authors with a *pro* stance and 173 authors with a *con* stance. A comparison of these predictions with the actual author stances is shown in Table 8.2.

If I put aside the authors for which the human judges could not determine the

Table 8.2: Author Stance classifier - Confusion matrix for dev set

	Predictions		
Actual	Pro	Con	Total
Pro	171	33	204
Con	22	107	129
Can't say	38	34	72
Total	231	173	405

actual stance, we are left with a resulting set of 333 authors. As can be seen from Table 8.2, the model made the correct prediction in 278 of these cases, which amounts to an overall accuracy of 83.5%. The corresponding macro-averaged F1-score is 82.9%. This is a significant improvement over a naive coin-flip baseline of 50%. The cross entropy error for this model was 0.214.

8.3.1.1 Accounting for the confidence level of component predictions

In this section, I experiment with different ways of factoring into the integrated author stance prediction model the associated confidence of the component comment topic stance and agreement classifier predictions. Intuitively, we would want to give greater weight to the higher confidence predictions, as these are more likely to be correct. On the one hand, to ensure the quality of the inputs to the integrated model we might wish to down-weight (or even ignore) the low confidence component predictions for the comment stance and agreement classifiers, since these could reflect comments or comment-response pairs that do not contain much information, and so have the potential of adding noise to the downstream task of author stance prediction. On the other hand, by only retaining instances for which we are very sure that we have the correct predicted comment or agreement label from the component classifier, we are potentially throwing away too much information that may be helpful in the author stance prediction task. For example, consistent, moderate predictions of a *pro* stance for comments written by the same person may provide just as much (if not more) information about the stance of this author than one or two higher confidence predictions would. Moreover, discarding lower confidence component predictions could also result in a subset of the authors now being out of reach of the author stance classifier, since they are no longer being represented in any of the

Table 8.3: Component classifiers - High confidence predictions for dev set

Component Classifier	Number of predictions	Authors covered
Comment Topic Stance (X_1)	4,235	324
Comment Pair Agreement (X_2)	1,671	156
Author Username (X_3)	34	34
Author Pair Agreement (X_4)	1,598	381
Total number of predictions	7,538	395

set of component classifer predictions. The goal is to find the right balance between the quality and the quantity of the classifier predictions, discarding the uninformative cases, resulting in a tighter, better performing model.[6]

I considered different retention boundaries for the comment stance and agreement classifier predictions, in terms of how close the scores were to 0 (equivalently, how close the components' predicted probabilities were to 0.5), in units of 0.1. In other words, I experimented with discarding predictions with scores in the range (-0.1, +0,1), (-0.2, +0,2), and so on. For each cut-off level I compared the resulting model classification accuracies, F1 scores, and cross entropy errors.

I found that for both the comment stance classifier and the agreement classifier, better overall results could be gained by ignoring the low confidence predictions with scores between -0.3 and +0.3 (corresponding to predicted probabilities in the range 0.35 to 0.65). For the comment stance classifier this meant discarding the predictions for some 2,102 comments, representing 33.2% of the data. There was a corresponding reduction of 3,168 (65.5%) of the agreement classifier predictions, from 4,839 down to 1,671. This latter result reflects the fact that the majority of comment-pairs are neutral in that they do not contain information that indicates the level of agreement between the two comments. The resulting set of high confidence predictions is summarized in Table 8.3.

As a result of discarding the lower confidence comment topic stance and agreement predictions in this way, there are a number of authors (38) who no longer receive a prediction from the overall stance prediction model, since none of the component classifier inputs to the model now relate to comments written by them. Moreover, the model is

[6]A pleasing side effect is that, with fewer variables the optimization algorithm converges more quickly.

Table 8.4: Author Stance classifier - High confidence predictions for dev set

	Predictions		
Actual	Pro	Con	Total
Pro	173	31	204
Con	21	108	129
Can't say	41	31	72
Total	235	170	405

not able to assign a stance to a further four authors, since these are two author-pairs who are in isolated 'agreement islands', meaning that although the model has some certainty about whether the author pair agreed or disagreed in their overall stance, there was no principled way to pin this to a *pro* or *con* orientation. For these 42 authors, the model determines the author stance based on a coin flip. Given this reduced, higher confidence, input dataset, the model predicted a total of 235 authors with a *pro* stance and 170 authors with a *con* stance. A comparison of these predictions with the actual author stances is shown in Table 8.4.

These results show that discarding the lower confidence predictions from the comment stance and agreement classifier components results in a slight increase in model performance. Classification accuracy increases from 83.5% to 84.4%, and the F1 score rises slightly from 82.9% to 83.8%. More importantly, however, the cross entropy error reduces 12.2%, from 0.214 to 0.188, meaning that the model is more confident in its predictions of author stances, validating the methodological decision to ignore the lower confidence predictions.

In the sections that follow, I examine the relative importance of each of the component classifiers to the overall model performance. I show the model results that arise if the predictions of each classifier are left out in the accumulation of the total cost function, and compare the resulting author stance predictions with those from the high confidence variant of the full model.

8.3.1.2 Contribution of Comment topic stance classifier - X_1

Without even running the calculations, it should be obvious that not using the information given by the comment topic stance classifier would have a serious deleterious

effect on the performance of the author stance prediction model. This is because by throwing away the 4,235 high confidence predictions from this component classifier, we lose vital evidence in respect of the stance of some 324 of the discourse participants. Even though many of these authors are also represented in the predictions of the agreement classifier, this would only allow us to determine whether they are on the same or opposite sides of the debate as each other; it would not automatically allow us to decide which side, *pro* or *con*, each individual author is on, and the polarity of the stance would be decided by a coin flip. The only hooks that would remain are those available from the predictions of the username classifier for a small number of authors. These predictions when combined with the agreement classifier and author alignment predictions would allow for stance predictions to be made for a much smaller set of authors than if the comment stance classifier predictions were included in the full model. The expectations for the model performance would therefore be significantly lower.

Indeed, this is precisely what I found: excluding the contribution of the comment topic stance classifier predictions X_1 from the cost function Z results in significantly poorer model results. Classification accuracy drops dramatically from 84.4% to 56.2%, and the F1 measure falls from 83.8% to 55.5%, just slightly greater than a random baseline. This confirms the relative importance of accurate comment stance detection in the task of author stance classification.

Coming at it from the other direction, now consider the results if the author stance prediction model was run using *only* the predictions of the comment topic stance classifier in the accumulation of the cost function, and ignoring the contribution from the agreement and other classifiers. Under this scenario, I can only expect to get model predictions for 324 of the authors, and the remaining 81 cases would be determined randomly. Here, I find model achieves a classification accuracy of only 74.8% (and a F1 score of 73.8%). This is 9.6% below the full model (i.e. including all components) results described in the previous section, and this quantifies the value of including other sources of information (from agreement, usernames and discourse structure) in the task of author stance prediction than using the text of the comments alone.

Finally, it is also informative to look at how well the classifier performs at the micro task of classifying the comment stance of the posts in the development set. Recall

that this classifier was trained on an external dataset, namely on a set of comments collected from two debating websites, www.debate.org and www.procon.org, on the topic of marriage equality. Although it was the same discussion topic of marriage equality underlying the comments in the training set and the development set, it is possible that there are some subtle differences between the data in those debating forums and the discussions on www.politico.com, that are limiting the performance of the model.

To be sure, we do not have ground truth labels of the topic stance for all of the 6,337 comments in the development set, and so it is not possible to carry out a direct evaluation of the comment topic stance classifier as it applies to these data. However, we do have the gold standard annotations of the stances of the authors of these comments. If we make the simplifying assumption that every comment written by author a_i will have the same stance - that being the underlying stance of author a_i - then we do have (noisy) labels for the set of comments based on which we can evaluate the component classifier. Of the comments in the development set, 3,829 were written by the 204 authors that the human annotators judged to be *pro* marriage, and 2,102 comments were written by the 129 *con* authors in the discussion. We cannot get labels for the remaining 406 comments, as these were written by authors whose stance could not be determined.

Applying the comment topic stance classifier to this set of 5,931 comments, we find the model achieves a classification accuracy of 68.9%, and a corresponding F1 score of 67.9%, indicating that the model was able to detect some signal in the data. This is only slightly below the level of performance of the classifier as it was applied to the held-out in-domain test data, as described in Chapter 5. To recall, those results showed 70.4% classification accuracy and a F1 score of 69.4%. This vindicates the methodological decision to train the comment stance classifier on an external dataset. The relatively small reduction in performance suggests that the potential issue of domain transfer was not significant: the www.politico.com data were not too dissimilar from the debating websites comments.

8.3.1.3 Contribution of Agreement classifier - X_2

To assess the importance of automatic agreement detection to the overall task of author stance prediction, I reran the model excluding the 1,671 predictions of the

agreement classifier in the development of the cost function. The resulting classification accuracy falls slightly from 84.4% to 81.1%, and the F1 measure falls 3.6 percentage points from 83.8% to 80.2%. At first glance, this appear to disconfirm the original intuition that cues about the level of agreement between pairs of adjacent comments in the discourse can fill in critical gaps that are left by only looking at the text of a comment to predict the stance of the author of that post. However, recall that this result still includes the contribution of the author pair alignment model component, X_4, which predicts that pairs of commenters who interacted in the discussion generally disagreed with each other. Indeed, if we omit both the agreement classifier predictions X_2 as well as the author pair alignment predictions X_4, the resulting model performance drops significantly to 75.9% accuracy (75.0% F1 score). This suggests that these two components together do provide important information about the interactions between authors in the discourse that is crucial for the task of determining the stance for every author.

8.3.1.4 Contribution of Username classifier - X_3

I have already shown in Chapter 7 that the username classifier is highly accurate, predicting the correct orientation of the set of 34 authors containing an orientation term in their username in 88.2% of cases. However, I find that in the overall author stance prediction task, this classifier does not make a significant contribution. Omitting X_3 from the calculation of the cost function results in only a small reduction in F1 measure (from 83.8% down just 0.4 percentage points, to 83.4%, which is not statistically significant). This is perhaps not too surprising given that the username classifier provided information on just 8% of the total authors in the discussion. Analysis of the results indicated that the authors who do have a username that reflects their ideological worldview were generally more active discourse participants than the average commenter in the discussion. Consequently, there were a greater number of individual comments and comment-response pairs involving these authors, on which the other component classifiers were able to make predictions. What is more, inspection of these comments suggested that they generally were more obvious as being on one side of the debate or the other, and so the classifier would have an easier job with them. In short, the username classifier provided information that was generally consistent with the predictions of the comment topic stance classifier,

and so is in some sense redundant. However, including the username classifier in the full model did allow for more confident predictions overall. I can see this by noticing that the cross entropy error relating to the reduced model without the username classifier (0.198) was around 5% lower than that for the full model (0.188).

8.3.1.5 Contribution of Author pair alignment model - X_4

I also showed in Chapter 7 that the model which predicts the likely ideological alignment between two discourse participants based on the number of times they interact in the dicussion is also highly accurate, predicting the correct agreement status in 1,380 out of 1,414 cases in the development set. In terms of the contribution this component makes to the overall author stance prediction task, I found that removing this piece results in a small reduction in F1 score from 83.8% to 80.3%. This is almost identical to the reduction in model performance if the agreement classifier component were omitted, as described in 8.3.1.3. This suggests that these two components are contributing consistent and potentially duplicative information to the model. However, including both components results in lower cross entropy loss (0.188, compared to a loss of 0.204 if the author pair alignment component were omitted) and therefore more confident predictions.

8.3.1.6 Discussion of component contributions

In the previous sections, I described the contribution of each of the four components towards the overall task of author stance detection in the development dataset. A summary of the results is shown in Table 8.5. I showed that all of the classifiers had a net positive contribution, either by increasing the classification accuracy or F1 score (the comment stance classifier and the agreement classifier) or by increasing the confidence of predictions and reducing the cross entropy error loss (the username classifier and the clues to author pair alignment given by the discourse structure).

Overall, the username classifier had the highest precision of the components, but the lowest recall, in that it only made predictions in a small number of cases. While the comment topic stance classifier had lower accuracy per se, it was the biggest overall contributor to the model. This finding indicates that future efforts to improve the author stance prediction model would be best spent improving the component comment stance

Table 8.5: Author Stance classifier - Relative contributions of component classifiers

Component Classifiers	Acc (%)	F1 (%)	CE Error	Description
X_1	74.8	73.8		Comment stance only
$X_1 + X_3$	75.9	75.0		Comment stance and usernames
$\varnothing + X_2 + X_3 + X_4$	56.2	55.5	0.344	No comment stance
$X_1 + \varnothing + X_3 + X_4$	81.1	80.2	0.210	No agreement
$X_1 + X_2 + \varnothing + X_4$	84.0	83.4	0.198	No usernames
$X_1 + X_2 + X_3 + \varnothing$	81.3	80.3	0.204	No author pair alignment
$X_1 + X_2 + X_3 + X_4$	**84.4**	**83.8**	**0.188**	High confidence predictions
$X_1 + X_2 + X_3 + X_4$	83.5	82.9	0.214	All predictions

classifier rather than any of the other components.

I also experimented with switching out the agreement classifier predictions with a more naive component model that blindly predicts a label of disagreement for every comment-response pair regardless of the textual content of the post or the reply. This is functionally equivalent to using a naive simplification of the discourse structure author alignment model (X_4), in which any attested author pair in the discussion is automatically given a predicted probability of agreement equal to zero. I found that using this naive agreement component instead of the existing agreement classifier (X_2) and discourse structural clues (X_4) results in lower F1 score (79.0%), which is greater than achieved by leveraging comment topic stances alone (73.8%), but less than using the richer information from the agreement classifier. Specifically, the automatic detection of positive agreement (as opposed to disagreement) between adjacent posts does improve the performance of the model.

Finally, we should consider the potential issue of *domain adaptation*, given that the two main text classifiers were trained on source data that was not from discussions taking place on the politico.com website, albeit that the training datasets are within the same overall genre of internet discourse, and - for the comment topic stance classifier - are on the same topic of marriage equality. Fortunately, this does not appear to be a major factor. The performance of the agreement classifier performs comparably on source and target data sets. The comment topic stance classifier performs somewhat worse on the target data, however, given the relatively small size of the training set, I do not consider

this to be a huge cause for concern. Nevertheless, further investigation of this matter is probably warranted.

8.3.2 Test Set results

I now apply the author stance classification model to the test set. Recall from Section 2.1.3 that the test dataset is similar to the development dataset, in that it relates to an online discussion about marriage equality on www.politico.com, two months later. It was a larger discussion than the development dataset, with a total of 7,755 comments made by 623 unique authors (excluding those with a username of *Guest*). The human judgments of the stances of these authors, and the results were 407 authors were judged to be in favor of same-sex marriage (or having liberal values in general), 127 against, and for the remaining 89 authors it was not possible to judges to determine their stance with certainty. The ratio of liberal to conservative commenters in this dataset was higher than in the development set.

Given that the development and test datasets were discussions on the same topic on the same platform just a couple of months apart, it would not be surprising to find that some commenters who participated in the earlier discussion also appeared in the later one. In fact, as Chapter 2 explains, there were 96 such authors in common between the development and the test datasets. The posts made by these 'returning' authors in the latter discussion were entirely new contributions to a different discourse with different interlocutors, and so there is no reason to assume that they would be repeating the very same patterns of language use across the two discussions to express their stance on the topic. Other than the username associated with the comments, no information learned from the development set for this set of commenters was leveraged in the evaluation of the model performance on the test set. In fact, this group of commenters provides another way to assess the stability and generalizability of the model, as - assuming that their actual stances did not change over a two month period (which is not an unreasonable assumption, per the Pew Research Center (2016) survey) - we would hope to see consistent model predictions for these authors across the two discussions. This turns out to be the case, as I will discuss further in the results section below,

I ran the comments, comment-response pairs, usernames, and discourse structure

Table 8.6: Component classifiers - Predictions for test set

Component Classifier	Number of predictions	Authors covered
Comment Topic Stance (X_1)	4,575	421
Comment Pair Agreement (X_2)	4,480	333
Author Username (X_3)	51	51
Author Pair Agreement (X_4)	2,060	565
Total number of predictions	11,166	570

through the four relevant component classifiers, and a summary of the predictions in respect of this dataset is as shown in Table 8.6. Note that the number of predictions for the comment topic stance and the agreement classifiers reflect just the high confidence predictions, based on the cut-off parameter of 0.3 tuned on the development set. As before, removing the lower confidence predictions for the text classifiers results in some authors not being represented in any of the predictions of the component classifiers, or whose stance polarity could not be determined by the model. For the test set, this amounted to a set 77 authors, and for these cases the model predicts the author stance by flipping a coin.

Given the inputs detailed in Table 8.6, the model predicted a total of 402 authors with a *pro* stance and 221 authors with a *con* stance. A comparison of these predictions with the actual author stances is shown in Table 8.7. Setting aside the 89 authors for which the human judges could not reliably determine the actual stance, we are left with a resulting set of 534 authors. As can be seen from Table 8.7, the model made the correct prediction in 426 of these cases, which amounts to an overall accuracy of 79.8%. The corresponding F1 score is 75.5%, representing the greater imbalance between the classes in the test set compared to the development set. This is a significant improvement over a naive coin-flip baseline of 50%.

As we might expect, the results on the test set are lower than those for the development set. This is because certain parameters - namely the cut-off for high confidence predictions and the coefficients for the regression in the author-pair alignment model - were derived from experimenting with the development data. Nevertheless, the test set results are significantly greater than baseline, and suggest that the model trained on exter-

Table 8.7: Author Stance classifier - Confusion matrix for test set

	Predictions		
Actual	Pro	Con	Total
Pro	325	82	407
Con	26	101	127
Can't say	51	38	89
Total	402	221	623

nal data and tuned on the development set is generally applicable for this and potentially also future cases on online discussions on this topic.

It is also informative to look at the model's predictions on different slices of the author data, according to how many comments each author posted in the discussion. As described in Chapter 2, the number of comments left by authors in the development and test sets followed a typical power law distribution, with a small number of commenters writing a large number of posts, and a very long tail of authors leaving just one or two comments. We would expect that the model would do a much better job at predicting the stances of the heavier commenters, given that there is more data from these authors to send to the classifiers. Moreover, given the potential applications of an author stance prediction model, it may be more useful to accurately detect the stance of the power users as well as the rump of commenters who participate moderately, rather the infrequent, occasional posters.

Table 8.8 shows the model predictions of author stances broken out by the quartiles of authors (based on the number of comments the authors posted in the discussion), as well as the classification accuracy and F1 score associated with each of these quartiles. First, notice the highly skewed distribution of the data. The top 25% commenters between them posted a total of 5,843 comments (78.9% of the total posts in the discussion), which is an average of 37.5 comments each. At the other extreme, every one of the 155 authors in the lowest frequency quartile each posted just a single comment to the discussion, and in aggregate contributed only 2.0% of the overall comments.

Clearly, there is a great deal more data in respect of the more frequent commenters, and therefore we would expect the model's predictions to be more accurate for these cases. This is also reflected in the ground truth author stance labels. The human judges clearly

Table 8.8: Author Stance classifier - Results for test set by frequency

			Actual			Predicted			
	Posts	Authors	Pro	Con	Uns	Pro	Con	Acc%	F1
Quartile 1	5,843	156	100	49	7	104	52	89.8	88.7
Quartile 2	1,408	156	109	32	15	106	50	82.0	78.4
Quartile 3	349	156	110	23	22	103	53	77.5	69.8
Quartile 4	155	155	88	23	45	89	66	65.2	59.0
Total	7,755	623	407	127	89	402	221	79.8	75.5

had a harder time determining the actual stances for the lower participating authors, as evidenced by the increasing number of 'unsure' cases as the level of prolificness of the author reduced. For the most frequent quartile of commenters, there were only 7 (4.5%) authors for which the annotators could not agree on a topic stance, whereas this percentage is as high as 29% for the bottom quartile of authors.

As to be expected, we can see from Table 8.8 that the model did much better predicting the stances for the most frequent authors, with an overall classification accuracy of almost 90% (and a corresponding F1 score of 89%) for the highest participating commenters. This is because the model has more inputs to work with, with respect to these authors: a greater number of high confidence predictions from the comment topic stance and agreement classifiers, and more instances of interactions with other discourse participants which the author pair alignment model can utilize. For many of the lowest quartile participants, on the other hand, there were few (if any) inputs from the component classifiers, and so the model's predictions were based on a random baseline, and the classification accuracy drops to 65%.

I close this section with a final observation of the model results. In Chapter 2, I explained that there were some 96 authors in common between the development and test datasets, and that 82 of these authors had consistent human judgments of their ground truth author stance across both discussions. This subgroup of discourse participants consists of 60 liberal and 22 conservative authors. I notice that 63 of these authors fall into the top quartile of most prolific commenters in the test set, and all but three of the remainder are in the second quartile. This is not a surprising result given that one may expect a high degree of correlation between the degree of active participation in the

www.politico.com commenting community as measured by the number of discussions an author is involved in (particularly, multiple discussion on the same topic), and the number of comments he or she leaves in each of those discussions. What this does mean is that these are cases for which the model has more, and more reliable, data, and so we should expect more accurate predictions of those authors' stances.

This feature of the test dataset raises two questions: (i) does the model perform better on the subgroup of 'returning' commenters in the test set than on the other commenters (which might be an indication of some sort of overfitting)?, and (ii) how consistently does the model predict the stances for the 'returning' authors across the two discussions, given that the predictions were generated independently, i.e. not using the data from one discussion to influence the stance predictions of the other discussion?

On the first question, I found there was essentially no difference at all in the classification accuracy for the subgroup of 96 'returning' commenters (80.7%) compared to that for the remaining set of 527 'new' authors (79.4%). This is not surprising given that the development set is not being used for learning the parameters of the model (other than for two hyper-parameters); as you recall, the core component classifiers (the comment topic stance, agreeement and username classifiers) were trained on external datasets.

On the second question, the model predictions for this set of 82 authors was found to be remarkably consistent across the two data sets: the predicted stances based on the test dataset matched the predictions from the development dataset in 69 cases, or 84% of the time, significantly greater than would be expected by chance. The replication of the predicted stance for a large group of commenters provides more evidence for the legitimacy of the model, in that it is able to detect a true signal in the underlying data.

8.4 Discussion

In the previous section I showed how an integrated model that takes inputs from various component classifiers to jointly classify the stance of commenters in an online discussion performs better than a simpler model that only considers a single perspective on the data, such the text of a comment taken in isolation. For example, an agreement classifier can provide information about the relationship between two authors that is

critical for determining the stances of those two individuals, which could not otherwise be gleaned from a comment stance classifer alone. Moreover, the accumulation of evidence from different sources - when this evidence is consistent and self-reinforcing - can serve to increase the confidence of the overall model predictions, as reflected in models with lower cross entropy.

When the pieces of evidence from the different component classifiers are not completely self-consistent, the joint model will weigh the various inputs together, and will determine the collective classification of author stances that is most consistent with the available evidence. In doing so, the main model implicitly allows for some of the component classifier predictions to conspire together to overrule other component predictions. For example, if a single individual prediction from the comment stance classifier, say, indicated that the comment had a *con* stance, yet all other predictions from this classifier for comments written by the same author indicated the opposite (or if this stance was inconsistent with predictions from the agreement or username classifier classifier), then the final *pro* author stance prediction given by the joint model would contradict the polarity of that original prediction, effectively overturning it. It turns out that we can get additional insights into the data by analyzing the component classifier predictions that are overturned by the preponderance of evidence given by other component predictions.

In many cases, to be sure, an individual component prediction is incorrect because of noisy data or because the underlying component model is not 100% accurate. This problem is mitigated to some degree by only retaining the higher-confidence predictions of the comment topic stance and the agreement classifiers. Nevertheless such misclassification errors will remain and these will get propagated to the collective classifier, where they will hopefully be caught and corrected.

However, if there are consistent patterns for a given author in the discourse, or an unusually high number of component predictions being overturned, this may be evidence of some other underlying phenomenon at play. In this section, I briefly discuss a couple of these cases: (i) a spot check on the quality of the data, specifically the ability to pin-point potentially non-unique usernames in the data set, and (ii) the identification of 'chaos-creaters' or 'trolls' - discourse participants whose sole purpose in the discussion appears to be fomenting disputes with others.

Detection of duplicate usernames

An analysis of the overturned component classifier predictions was critical in this research in the detection of an anomaly in the development dataset. In this data, I noticed that one author named *Skeptic* had posted 95 separate comments (thereby landing in the top 10% of most-frequent commenters), with 78 attested interactions with 48 other discourse participants (either by responding to somebody else's posts, or by receiving replies). The comment topic stance classifier returned 37 high-confidence predictions in respect of *Skeptic*'s 95 comments, and these were split pretty evenly, with 17 comments predicted to be *pro* and 20 comments predicted to be *con*. The agreement classifier returned 35 high-confidence predictions with respect to his or her 78 cases of interactions with other discourse participants.

Using these component predictions as inputs to the author stance prediction model, *Skeptic* was determined to have a *pro* stance (with an associated predicted probability of 0.71), meaning that 20 of the component comment topic stance classifier predictions were incorrect. Moreover, taking account of the predicted stances for the other authors *Skeptic* interacted with, some 18 of the component agreement classifier predictions were also deemed to be incorrect. The corresponding ratios of the number of incorrect to total predictions from these two component classifiers was much higher than for other authors with similar levels of participation in the discourse, and so warranted a closer inspection of the data.

Looking at the highest confidence predictions of the comment topic stance classifier, I found a number of comments that were very clearly written by an author with a strong anti-gay stance, as in (12a), whereas other comments seemed to be coming from the pen of a liberal commenter, as in (12b). None of these cases was an obvious example of sarcasm.

(12) a. **Skeptic:** Selfish Gay Infant Narcissists: "Bake me a cake NOW!"

b. **Skeptic:** We used to have trolls who could at least hold up an argument. Now they just cut-and-paste the same garbage to filibuster the board. This is why you lost the culture way, guys.

I also checked the human judgments of the ground truth of the stance of *Skeptic*. In the original annotation file, I saw that this was a case where two of the three annotators

indicated that it was not possible to infer the stance, and the third annotator judged the stance to be conservative. This was consistent with the observation of the data.

I returned to the www.politico.com website itself to double check that no errors had crept in during the process of scraping the data via the API. In doing so I noticed that the comments posted by *Skeptic* were associated with two different avatars (thumbnail photos) in the comment threads. Digging further, I was able to conclude that there were indeed two separate Disqus users with the handle *Skeptic*, and both had participated in the marriage equality discussions on www.politico.com. It turns out that the username displayed above a post on the website (and the name associated with the comment in the information returned by the API), is not in fact unique. In most cases, this name matches the unique login ID that a user creates when setting up her or his account on Disqus. However, users have the option to change this 'display name' to something else - and that is clearly was the one of the *Skeptic*s must have done. This explains why this author's comments were so internally inconsistent, and why the human judges were not able to agree upon a stance for this commenter.

I was eventually able to access the unique name identifier associated with each commenter in the discussion, and partition the set of *Skeptic*'s 95 comments into two sets of 65 and 30 comments, respectively, that were posted by the two different users. I was also able to confirm that there were no other cases of duplicate usernames represented in the dataset. All of the results shown and discussed in this chapter reflect the cleaning up of the dataset and ground truth annotations to account for the two *Skeptic*s.

I realize that this is anecdotal evidence, and - now that the issue had been discovered and addressed - the ability to detect duplicate usernames is not something that will be necessary going forward in the analysis of other datasets gathered from this source. However, the case of the two *Skeptic*s does provide some validity to the underlying model methodology whereby authors stances are jointly inferred to maximize the likelihood of all of the available evidence.

Detection of chaos creaters

In this genre of online debate, there inevitably seems to be a small handful of discourse participants in any discussion who are not engaging in a good faith debate on the topic at

hand, but instead whose sole purpose appears to be to post inflammatory, devil's advocate, or off-topic comments, or simply to get into verbal disputes with others just for the sake of it. These chaos creaters diminish the quality of the debate, and can hijack discussion threads that would otherwise be places of reasoned debate (albeit, sometimes heated and disputative) between discourse contributors who adhere more strictly to Gricean norms of communication. If it were possible to automatically detect these intentionally disruptive commenters, commenting platforms could be configured to hide or deprioritize their posts, and thereby maintain the overall quality of the discussion. I suggest that the author stance model presented in this chapter, coupled with an analysis of the overturned component predictions, provides a small toe-hold towards solving this problem. I sketch here how this could possibly work, but leave a more robust development of these ideas to future work.

I start by going back to the gold standard annotations of author stances for the development dataset, described in Chapter 2. As you recall, there were a total of 72 authors (out of a total of 405) for whom the human judges were not able to agree upon a firm topic stance. As discussed in that chapter, many of these cases simply related to the fact that there was not enough evidence available, as the author had posted just a single comment or two, or that the comment texts themselves were vague or off-topic. However, if we focus just on the top quartile of the most frequent contributors, we find that there are three authors - namely, *John*, *Jeff Cigar*, and *lalameda* - who despite their having posted an average of 54 comments each to the discussion, the human judges could not determine their topic stance. It is precisely these cases that it would be useful to be able to detect.

I analyzed the component classifier predictions in respect of these three authors, as well as the output of the integrated author stance prediction model, and found that the data had the following characteristics:

- A high number of interactions with other discourse participants
- A low number of (high confidence) comment topic stance classifier predictions, X_1
- A high number of (high confidence) agreement classifier predictions, X_2, with the majority of these being (i) where the author's post was a reply to a previous com-

ment, and (ii) the polarity of the prediction was negative (i.e. disagreement)

- An overall author stance prediction with low confidence
- A high ratio of the number cases where the component classifier predictions were overruled by the overall author stance prediction to the number of predictions

Consider the author *John* as an example. This author had posted a total 46 comments in the discussion, interacting with 22 other authors across the discourse. Few of this author's posts gave a strong indication of his stance on the topic of marriage equality, and the comment topic stance classifier only provided five predictions with a confidence greater than the established cut-off, which represents only 10% of his comments. By way of comparison, the corresponding average for the group of the 101 most frequent commenters is 28%. The agreement classifier on the other hand returned 45 high confidence predictions with respect to comment-pairs where *John* was one of the authors involved, out of 58 possible comment-pairs. Of these cases, 35 were instances where *John* had written the reply comment, rather than having one of his comments replied to. The ratio of 'responding' to 'being-responding-to' is much closer to 50:50 for the entire set of the 101 most prolific authors. This indicates that *John* is much more reactive than the other authors, on average. Moreover, 43 of these 45 agreement classifier predictions were cases of predicted disagreement (as opposed to predicted agreement). Again, this ratio is much higher than the average 'predicted disagree' to 'total' for the group, which is a little greater than 80%.

The author *John* appears to be disagreeing with other discourse participants across the board, even with other authors who have the opposite stance as each other on the issue of marriage equality. Given these self-contradictory pieces of evidence being provided from the component classifiers, the integrated author stance model has a hard time handling these inconsistent inputs. The overall (binary) author stance prediction for *John* given by the model is *con*, but the associated real-valued predicted stance score was only slightly negative, indicating a low confidence of prediction. Moreover, assuming a *con* stance for the author would mean that of the 72 total high confidence predictions given by the component classifiers, some 31 of them (43%) would be directionally incorrect. For instance, the model determines that many of the comment-pairs to which John contributed

should really be cases of agreement, since it infers a *con* stance for both of the authors based upon the totality of the evidence. However, this contradicts the prediction of the agreement classifier for these cases. This percentage of overturned predictions is greater than the level of 34% observed for the group as a whole.

The question is, are these observations generalizable? Would it be possible to crunch these same statistics over other discussions - such as the test dataset used in this dissertation - to be able to identify authors like *John* and *Jeff Cigar* in those debates? In that test dataset, I found that there are exactly seven 'top quartile' authors (i.e. the 156 most frequent posters in that discussion) for whom the human annotators could not agree on topic stance. For this data set, I took the results of the component classifier predictions along with the integrated author stance model predictions, and calculated the following metrics for each of the top quartile discussion participants: (i) the ratio of the number of high confidence comment topic stance classifier predictions to the number of comments posted; (ii) the fraction of the comment-response pairs involving the author which were cases where the author provided the response; (iii) the ratio of the number of cases of predicted disagreement to the number of high confidence agreement classifier predictions pertaining to the author; (iv) the level of confidence of the author stance prediction; and (v) the proportion of the component classifier predictions that were contradicted by the overall author stance determined by the model.

Rather than constructing another classifier, instead for now, I use basic heuristics to investigate whether these metrics are indicative of discussion participants that could be considered chaos creaters. I sort each of the five lists of values into decreasing order, and then look to see which authors appear consistently in the top n-th percentile of each list. I found that no author appears in the top decile of all five of these lists, but two authors (*korvu* and *spenser*) do appear in the top quintile (20%) of every list. The first of these authors is one of the cases of the seven for whom there is no gold standard stance. If we are less strict, and instead look for authors who appear in the top quintile of four of the five lists, we catch a total of twelve authors, including four of the seven potential chaos creators.

There is obviously a lot work to be done developing this - especially to mitigate the false positive cases - but it tentatively seems that there could be preliminary evidence

that the predictions of the component classifiers and integrate author stance model could be used to identify such authors in a discussion.

8.5 Conclusion

In this chapter I have presented a model that collectively classifies the topic stance (i.e. in favor or against a controversial topic) for every discourse participant in a particular online discussion, using component classifier prediction inputs that relate to the text of an author's posts, his or her interactions with other interlocutors, as well as metalinguistic information such as the tree-like structure of the discourse itself, and hint of the author's self-identification available from their user profile within the commenting technology platform. I showed that each component contributes positively to the overall model performance, and omitted that source of evidence would result in lower model accuracy and a reduction in the confidence of the predictions of the model.

Part III

Conclusion and Future Work

Chapter 9

Conclusion and Future Work

In this thesis I investigated the rich genre of multi-party, multi-threaded discourse that manifests in online discussion forums. I addressed the research question of whether it is possible to automatically detect the topic stance of commenters based upon their contributions to a discussion, and their interactions with other discourse participants - specifically in the case of online discussions on a polarizing ideological topic such as gun control or marriage equality. In this short conclusion, I describe the major contributions of this thesis and summarize the major findings. I then discuss some of the limitations of the work, and describe the future directions in which this research could be taken.

9.1 Summary of Contribution

I have presented a novel way of addressing the stance detection task utilizing information from model components built and trained to detect specific features of the discourse, and predicting a final stance for each author in the discussion that is most consistent with the evidence provided by these components. The four component classifiers were: (i) a comment topic stance classifier, used to detect the polarity of stance of an individual post taken out its discourse context; (ii) an agreement classifier, used to detect the level of disagreement between two adjacent posts in the discussion; (iii) a username classifier, for predicting the ideological orientation of an author based on the chosen screen name; and (iv) a regression model to predict the probability of overall agreement between any pair of two authors in the discussion.

The comment topic stance classifier presented in Chapter 5 was trained on a custom-built corpus that I developed by pulling data from two websites for debating. The design of the interface of these sites effectively provided the stance labels of debate posts automatically, without the need for costly human annotation. The classification model experimented with novel features that aimed to capture the propositional content of the underlying posts, which were shown to have predictive power, and the model achieved classification accuracy of 70%. The agreement classifier presented in Chapter 6 was trained on the publicly available Internet Argument Corpus (Walker et al., 2012b) and included features that had not been used in previous work involving this dataset. In Chapter 7, I identified two aspects of online discourse that have not before been used in models for stance prediction. First, I presented a username classifier that predicts the underlying ideological orientation of a user on the Disqus commenting platform. This component involved non-trivial subtask of handling idiosyncratic orthography and parsing a username without whitespaces into its composite parts. The username classifier achieved classification accuracy of 83% over a test set of data with ground truth labels determined by human judges. In the same chapter, I also presented an author-pair alignment model, which predicts the probability of ideological agreement between two discussion participants based solely on the discourse tree structure of the conversation and the number of instances of interactions between these two commenters over the course of the discussion. The fitted regression model produced predictions that were highly consistent with the observed data, with a root mean square error of 0.05, and an overall error rate of only 4%.

The final author stance prediction model presented in Chapter 8 was applied to a new corpus of annotated data that was collected for this work. The corpus comprised two separate data sets both involving discussions of marriage equality taken from the political website, www.politico.com. The data included the comment text of each posted contribution as well as the threaded structure of the discourse and associated metadata. The stances of the commenters participating in these discussions were annotated by a set of human judges, and used for the evaluation of the model. The model achieved a F1 score of 84% and 76% over the two datasets. The result for the second set of data was negatively impacted by the long tail of users in this discussion who only left a single

comment, thereby not giving the model much evidence to work with. The score for the upper quartile of authors based on the number of comments posted was 89%. I found that the comment topic stance and agreement classifiers were the most important components in predicting the polarity of the author stances. However, the additional information provided by the username classifier and the alignment regression model served to increase the confidence in the model predictions. The model also gave consistent predictions of the stances for the authors who happened to participate in both discussions, a result which provides an additional indication of the validity of the model.

9.2 Limitations and Future Directions

There are a number of limitations of the work presented in this theses, as well as many future potential directions in which the research could be extended or expanded. I summarize some of the main areas below.

9.2.1 Extending to other discussion topics and scaling up to larger data sets

A major shortcoming of the work presented in this thesis is its focus on two relatively-small data sets extracted from one source, politico.com, on the issue of marriage equality. While the model was able to discern the author stances of participants in these two discussions with a good degree of accuracy, further research is needed to determine whether similar results would be found if the model were applied to comparable data sets – either relating to different discussion topics, or from a different source, or both.

With respect to the matter of discussion topic, this question should not be too difficult to answer. First, the data collection process, via the Disqus API, would remain the same – the only requirement would be to identify the 'hot topic' news stories on socially controversial issues that would generate substantial discussion among commenters. Unfortunately, in today's hyper-polarized socio-political climate, these stories are all-too-commonplace. We would also need to collect more training data for the comment topic stance classifier. However, I have already discussed in Chapter 5 a process for doing just that, pulling self-labeled data from the two debating websites, procon.org and debate.com,

relating to the topic under discussion. The remaining component classifiers such as the agreement classifier should, theoretically, be usable as they are, since there is no principled reason why the language devices used to align oneself with respect to a prior comment in the discussion (such as praising, thanks, expressing agreement or disagreement, insulting, question assumptions, and so on) would depend upon the subject matter of the topic being discussed. Similarly, if the two stance positions on the topic map intuitively to a left-wing/right-wing dichotomy, the username classifier could also be utilized as-is with no need for modification.

Unfortunately, however, we quickly run into the issue of annotating the data with respect to the ground truth of the author topic stances for any new data set. For this thesis, I collected the human judgments of author stances for a total of over 1,000 commenters, who between them left a total of over 14,000 comments in two discussions. This was a complex and time consuming process, involving training a small team of research assistants, building a user-interface to facilitate the annotation process, and analyzing the annotations for consistency and to ensure quality. This annotation process followed here is simply not scalable if the project were to be extended to other topics in a non-trivial manner.

A direction of future research would be to explore ways to more easily attain labels for the true author stances of discussion participants. While crowdsourcing the task is an option, it is a lot more complicated than the typical Mechanical Turk job. This is because - as we have seen - it is oftentimes only possible to ascertain the stance of an author after looking at the aggregate of her comments over the course of the discussion, and in the contexts of the posts they were replying to. It is very difficult to atomize this into a collection of micro-classification tasks for a crowdsource worker.

A more promising line of research would be to find a way to use bootstrapping or distant supervision techniques to automatically infer the stance labels for unlabeled comments, using heuristics and a smaller, high-quality labeled data set. While these inferred labels inevitably contain a lot of noise, the sheer volume of new training data generated by distant supervision can more than outweigh this, often resulting in better models overall. The much larger training sets available in this way would also allow for more contemporary modeling choices for the component classifiers, such as the recurrent neural network

models that are prevalent in many current state-of-the-art natural language processing applications, but which were not suitable for the data in this thesis because of the limited size of these datasets.

9.2.2 Redefining the predicted class labels

The stance detection task at the center of this research was essential a binary classification problem: is author a_i fundamentally *in favor of* or *against* a given ideological topic? While this may be a valid question for truly polarizing social or politicial issues such as the topics like gun control and marriage equality discussed in this thesis, it is less clear if this remains the case for broader discourse topics, such as climate change, or more ethically open-ended questions, such as '*is it ever OK to steal*'? In these cases, people's stances are less easily categorized as falling neatly into just one of two buckets. Instead, more nuanced positions are often taken; people can express their support for one particular aspect of an issue, while indicating their skepticism about another. A more useful model, and one that would have more practical applications, would be one whether the categorical class labels were more sophisticated than a simple positive/negative distinction.

The simplest extension of the model to imagine would be to use a spectrum of labels, ranging from strongly and weakly negative through neutral to weakly and strongly positive, allowing for a more fine-grained prediction than a binary decision. Building on this, the class labels could be pairs of orthogonal characteristics, with political ideology along one dimension, say, and level of optimism about the future along the other, with the model making joint two-way predictions for each author. Yet another option would be hierarchical class categories, in which commenters are first categorized as belong to the left of the right of an issue like vegetarianism, and then further subcategorized according to the rationale they have for taking this position (such as health, animal welfare, the environment), and so on. Building and training models that are able to predict these more nuanced categories is obviously much more complicated than the task discussed in this thesis, requiring much more data, but the resulting models would likely have more practical application.

A related aspect that should be explored further was touched upon in the previous chapter in the discussion of chaos creators - participants whose only purpose in the

discourse seems to be to sow discord in the discussion, picking arguments with other commenter regardless of their true opinion on the topic. I presented a sketch of a methodology for how it might be possible to programmatically identify such non-cooperative discourse participants by looking at their patterns of interactions with other participants. It would be interesting to pursue this line of inquiry – particularly given the news over recent months about the involvement of Russian troll farms in US democracy and their participation in social media and in online discussion boards. The ability to identify fake news, and the people who spread it, is needed urgently.

9.2.3 Detecting sarcasm

As discussed at various times throughout this thesis, one of the defining characteristics of internet comments is the frequency of use of non-literal or ironic (or sarcastic) language. This sarcasm can be manifested in many ways. For one, it is commonly instantiated as a form of sarcastic praise (e.g. '*Now there's some persuasive debating.*') The surface form of a comment like this appears to be a positive sentiment directed at the author of the previous posts. An unsophisticated agreement classifier - given the presence of the positive sentiment terms - would likely classify it as an instance of agreement. In other cases, sarcasm is manifested as hyperbolic language (e.g. '*I think that arming teachers in schools is a GREAT idea - what could possibly go wrong?!?*'). It is obvious to human readers that this comment is not to be taken at face value, and in fact the stance of the author is likely to be precisely opposite to that implied by the literal interpretation. Such performative adoption of the language that could be used by a speaker on the other side of the debate would present a major challenge for a text classifier like the comment topic stance model. Indeed, the automatic detection of sarcasm is an extremely challenging proposition for any natural language processing task, and is still an underexplored research problem (see Joshi et al. (2017) for a recent survey).

The model presented in this thesis tries to overcome the effect of sarcasm to some degree by the way in which assesses each piece of evidence being passed along by the component classifiers in light of the other indicators it has. One or two non-literal comments posted by an author in a discussion would, one hopes, be counteracted by the accumulation of other evidence from the non-sarcastic posts. However, this approach would fail

if the majority of the comments written by an author do contain sarcasm (which is not that an unlikely a scenario for participants who post only once or twice over an entire discussion). It would be better to address the matter head on, and learn ways to detect the sarcasm directly.

Bibliography

Abbott, R., Walker, M., Anand, P., Fox Tree, J. E., Bowmani, R., and King, J. (2011). 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.

Abu-Jbara, A., Diab, M., Dasigi, P., and Radev, D. (2012). Subgroup detection in ideological discussions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*: *Long Papers - Volume 1*, pages 399–409. Association for Computational Linguistics.

Agrawal, R., Rajagopalan, S., Srikant, R., and Xu, Y. (2003). Mining newsgroups using networks arising from social behavior. In *Proceedings of the 12th International Conference on World Wide Web*, pages 529–535. ACM.

Anand, P., Walker, M., Abbott, R., Tree, J. E. F., Bowmani, R., and Minor, M. (2011). Cats rule and dogs drool!: Classifying stance in online debate. In *Proceedings of the 2nd Workshop on Computational Approaches to Subjectivity and Sentiment Analysis*, pages 1–9. Association for Computational Linguistics.

Bansal, M., Cardie, C., and Lee, L. (2008). The power of negative thinking: Exploiting label disagreement in the min-cut classification framework. In *COLING (Posters)*, pages 15–18.

Bechar-Israeli, H. (1995). From BONEHEAD to cLoNehEAd: Nicknames, play, and identity on Internet Relay Chat. *Journal of Computer-Mediated Communication*, 1(2).

Bender, E. M., Morgan, J. T., Oxley, M., Zachry, M., Hutchinson, B., Marin, A., Zhang, B., and Ostendorf, M. (2011). Annotating social acts: Authority claims and alignment moves in Wikipedia talk pages. In *Proceedings of the Workshop on Languages in Social Media*, pages 48–57. Association for Computational Linguistics.

Biran, O., Rosenthal, S., Andreas, J., McKeown, K., and Rambow, O. (2012). Detecting influencers in written online conversations. In *Proceedings of the Second Workshop on Language in Social Media*, pages 37–45. Association for Computational Linguistics.

Bird, S. (2006). NLTK: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics.

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

Brown, P. and Levinson, S. C. (1987). Politeness: Some universals in language usage, Volume 4. Cambridge University Press.

Byrd, R. H., Lu, P., Nocedal, J., and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190– 1208.

Danescu-Niculescu-Mizil, C., Lee, L., Pang, B., and Kleinberg, J. (2012). Echoes of power: Language effects and power differences in social interaction. In *Proceedings of the 21st International Conference on World Wide Web*, pages 699–708. ACM.

Davies, M. (2008). *The Corpus of Contemporary American English*. BYE, Brigham Young University.

Galley, M., McKeown, K., Hirschberg, J., and Shriberg, E. (2004). Identifying agreement and disagreement in conversational speech: Use of bayesian networks to model pragmatic dependencies. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 669. Association for Computational Linguistics.

Germesin, S. and Wilson, T. (2009). Agreement detection in multiparty conversation. In *Proceedings of the 2009 International Conference on Multimodal interfaces*, pages 7–14. ACM.

Giles, H., Coupland, J., and Coupland, N. (Eds.) (1991). *Contexts of accommodation: Developments in applied sociolinguistics*. Cambridge University Press.

Goodman, E. and Cherubini, F. (2013). Online comment moderation: Emerging best practices. *Germany: Darmstadt, The World Association of Newspapers WAN-*

IFRA. http://www.wan-ifra.org/reports/2013/10/04/online-commentmoderation-emerging-best-practices (17.9.2014).

Hasan, K. S. and Ng, V. (2013a). Extra-linguistic constraints on stance recognition in ideological debates. In *ACL* (2), pages 816–821.

Hasan, K. S. and Ng, V. (2013b). Stance classification of ideological debates: Data, models, features, and constraints. In *IJCNLP*, pages 1348–1356.

Hasan, K. S. and Ng, V. (2014). Why are you taking this stance? Identifying and classifying reasons in ideological debates. In *EMNLP*, pages 751–762.

Hassan, A., Abu-Jbara, A., and Radev, D. (2012). Detecting subgroups in online discussions by modeling positive and negative relations among participants. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 59–70. Association for Computational Linguistics.

Honnibal, M. and Johnson, M. (2015). An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Horn, L. (1989). *A natural history of negation*. Chicago, IL: University of Chicago Press.

Joshi, A., Bhattacharyya, P., and Carman, M. J. (2017). Automatic sarcasm detection: A survey. *ACM Computing Surveys (CSUR)*, 50(5):73.

Lindholm, L. (2013). The maxims of online nicknames. *Pragmatics of computer-mediated communication*, 9:437.

Liu, D. C. and Nocedal, J. (1989). On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45(1-3):503–528.

Lukin, S. and Walker, M. (2013). Really? Well. Apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. In *Proceedings of the Workshop on Language Analysis in Social Media*, pages 30–40.

Malouf, R. (2012). A comparison of algorithms for maximum entropy parameter estimation. In *Proceedings of the 6th Conference on Natural Language Learning Vol 20*, pages 1–7. Association for Computational Linguistics.

Malouf, R. and Mullen, T. (2008). Taking sides: User classification for informal online political discourse. *Internet Research*, 18(2):177–190.

Misra, A., Anand, P., Tree, J., and Walker, M. (2015). Using summarization to discover argument facets in online ideological dialog. In *NAACL HLT*, pages 430–440.

Misra, A. and Walker, M. A. (2013). Topic independent identification of agreement and disagreement in social media dialogue. In *Conference of the Special Interest Group on Discourse and Dialogue*, page 920.

Mohammad, S. M., Kiritchenko, S., Sobhani, P., Zhu, X., and Cherry, C. (2016a). Semeval-2016 task 6: Detecting stance in tweets. In *Proceedings of the International Workshop on Semantic Evaluation, SemEval*, volume 16.

Mohammad, S. M., Sobhani, P., and Kiritchenko, S. (2016b). Stance and sentiment in tweets. *arXiv preprint arXiv:1605.01655*.

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

Mukherjee, A. and Liu, B. (2013). Discovering user interactions in ideological discussions. In Association for Computational Linguistics (ACL).

Murakami, A. and Raymond, R. (2010). 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.

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

Pew Research Center (2016). When Social Media Changes Minds. URL: http://www.pewresearch.org/fact-tank/2016/11/07/social-media-causes-some-users-to-rethink-their-views-on-an-issue/ft_16-11-07_socialpolitics/. Accessed on March 22, 2018.

Pew Research Center (2017). Changing attitudes on gay marriage. URL: http://www.pewforum.org/fact-sheet/changing-attitudes-on-gay-marriage/. Accessed on June 30, 2017.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., and Webber, B. L. (2007). *The Penn Discourse Treebank 2.0 Annotation Manual*.

Qiu, M., Yang, L., and Jiang, J. (2013). Modeling interaction features for debate side clustering. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 873–878. ACM.

Razavi, A., Inkpen, D., Uritsky, S., and Matwin, S. (2010). Offensive language detection using multi-level classification. *Advances in Artificial Intelligence*, pages 16–27.

Rosenthal, S. and McKeown, K. (2015). I couldn't agree more: The role of conversational structure in agreement and disagreement detection in online discussions. In *16th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 168.

Schlöder, J. J. and Fernández, R. (2014). The role of polarity in inferring acceptance and rejection in dialogue. In *15th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 151.

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

Spertus, E. (1997). Smokey: Automatic recognition of hostile messages. In *AAAI/IAAI*, pages 1058–1065.

Sridhar, D., Getoor, L., and Walker, M. (2014). Collective stance classification of posts in online debate forums. *ACL 2014*, 109.

Stalnaker, R. C. (1978). *Assertion*. Wiley Online Library.

Stommel, W. (2007). Mein nick bin ich! Nicknames in a German forum on eating disorders. *J. Comp.-Med. Commun.*, 13(1):141–162.

Thelwall, M., Buckley, K., and Paltoglou, G. (2012). Sentiment strength detection for the social web. *Journal of the American Society for Information Science and Technology*, 63(1):163–173.

Thomas, M., Pang, B., and Lee, L. (2006). Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 327–335. Association for Computational Linguistics.

Walker, M. A. (1996). Inferring acceptance and rejection in dialog by default rules of inference. *Language and Speech*, 39(2-3):265–304.

Walker, M. A., Anand, P., Abbott, R., and Grant, R. (2012a). Stance classification using dialogic properties of persuasion. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 592–596. Association for Computational Linguistics.

Walker, M. A., Tree, J. E. F., Anand, P., Abbott, R., and King, J. (2012b). A corpus for research on deliberation and debate. In *LREC*, pages 812–817.

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

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). 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.

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