MINING AND EVALUATING ARGUMENTATIVE STRUCTURES IN USER COMMENTS IN ERULEMAKING

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Joonsuk Park

August 2016

MINING AND EVALUATING ARGUMENTATIVE STRUCTURES IN USER COMMENTS IN ERULEMAKING

Joonsuk Park, Ph.D.

Cornell University 2016

With the advancement of information technology, the amount of textual content written by inexperienced writers, in the form of comments, product reviews, blog posts, etc., is steeply rising. While it is a valuable medium of communication that enables people of various backgrounds to share their experiences and opinions, user generated text often consists of unclear argumentative structures and unsubstantiated claims, which renders it difficult for people to understand, let alone evaluate. This hampers effective communication and discourages productive discussion. It is especially problematic in eRulemaking, where the claims of the citizens are meaningful for government officials, and other citizens, only if they are adequately supported in a clear manner.

One approach to tackle this problem is to automatically extract and evaluate argumentative structures in user comments. This in turn would allow assistance at both ends of communication—guiding the writers to construct well-structured arguments through real-time feedback and reducing the burden on readers by means of summarization.

This thesis proposes a theoretical argumentation model to capture prevalent support structures in user comments and presents computational tools to extract from them components of the model. Argumentative structures can then be evaluated by comparing the expected type of support as defined in the model to the actual support provided by the writer, if any, in the given text.

BIOGRAPHICAL SKETCH

Joonsuk Park was born in Seoul in 1984. He graduated from Seorae Elementary School in 1997 and—after a semester at Bangbae Middle School—relocated to California. He graduated from Nicolas Junior High School in 1998 and John F. Kennedy High School—where he had his first exposure to computer science—in 2002. Following two years at the University of California at San Diego (UC San Diego), he returned to Seoul for a year long study at Yonsei University. He then served in the Republic of Korea (R.O.K.) Army as an interpreter, serving at the Korea Armed Forces Athletic Corps, a.k.a. *Sangmu*, in R.O.K. and the R.O.K. Armed Forces Support Group, a.k.a. *Dasan*, in Afghanistan from 2005 to 2007. After working as an English instructor and freelance translator, he returned to UC San Diego and graduated with a B.A. in Mathematics-Computer Science in 2009. Subsequently, he worked as a staff research associate at UC San Diego and a software engineer at NHN Corp.[1] Then, he joined the Department of Computer Science at Cornell University, where he completed his Ph.D. in 2016. While pursuing his Ph.D., he spent two summers abroad as a visiting researcher: Nara Advanced Institute of Science and Technology (NAIST) in Japan and the University of Dundee in Scotland, in 2013 and 2015, respectively. He began his appointment as a visiting assistant professor in the Department of Computer Science at Williams College in July 2016.

[1] At the time of writing, the official name of the company is *Naver Corp*.

To my parents.

ACKNOWLEDGEMENTS

Here is my vain attempt at recalling everyone that needs to be recognized:

• Claire Cardie, my **thesis advisor** and lifelong-mentor-to-be, for her guidance throughout my journey as a Ph.D. student. Her insights empowered me to see, and her patience and encouragements enabled me to breathe. I also thank the **committee members** Dan Cosley and John Hale for their advice regarding research and teaching. Looking back, I was fortunate to have a committee full of advisors who are not only competent researchers, but also passionate teachers.

• Yoav Artzi, Xilun Chen, Yejin Choi, Cristian Danescu-Niculescu-Mizil, Jack Hessel, Ozan Irsoy, Arzoo Katiyar, Lillian Lee, Moontae Lee, Parvaz Mahdabi, Dipendra Misra, Vlad Niculae, Myle Ott, Dinesh Puranam, Karthik Raman, Vikram Rao, Mats Rooth, Xanda Schofield, Ves Stoyanov, Chenhao Tan, Lu Wang, Bishan Yang, Ainur Yessenalina, and the rest of the past and present members of the **Cornell NLP gang** for the academic and social interactions.

• Abhishek Anand, Shuo Chen, Igor Labutov, Kisuh Lee, Albert Liu, Kevin Matzen, Amit Sharma, Jiyong Shin, Jaeyong Sung, Adith Swaminathan and others in the **Department of Computer Science** at Cornell for the fun and educational conversations. I especially thank Thorsten Joachims for taking me on as his student, which drew me to Cornell, and my student mentor, Brad Gulko, for his continued support throughout my graduate studies, literally from day 1.

• Cheryl Blake, Dmitry Epstein, Cynthia Farina, Josiah Heidt, Sally Klingel, Gilly Leshed, Brian McInnis, Elizabeth Murnane, Mary Newhart, Jackie Solovan, Joan-Josep Vallbe, and others at the **Cornell eRulemaking Initiative (CeRI)** for the interdisciplinary and social environment from which I have learned how to communicate and collaborate with experts from diverse fields.

• Colleen McLinn, David Way, Kimberly Williams and others at the **Center for**

Teaching Excellence (CTE) and the **Center for the Integration of Research, Teaching, and Learning (CIRTL)** at Cornell for practical lessons on pedagogy.

• Suwon Bae, Seungchun Bang, Inho Cho, Eunsol Choi, Seokhyun Jin, Laewoo Kang, Jaeyeon Kim, Jangwoo Kim, Jungyoon Kim, Kyuyoung Kim, Hankyul Kim, Seunghyun Kim, Haekyung Lee, Jimin Lee, Heehwa Min, Heyjun Park, Holim Jang, Hooyeon Lee, Joonyoung Kwak, Changhyuk Lee, Jungmi Lee, Mihae Lee, Junghee Park, Seongha Park, Myungjoo Shin, Younghye Song, Dongwook Yoon, and other members of the **Korean Graduate Student Association (KGSA)** at Cornell for the good times and the valuable leadership experience.

• Floris Bex, Katarzyna Budzynska, Rory Duthie, Mathilde Janier, Barbara Konat, John Lawrence, Chris Reed, Mark Snaith, and Olena Yaskorska at the **Centre for Argument Technology (ARG-tech)** at the University of Dundee for the lessons on argumentation with endless drinking and secondhand smoking!

• Nakamura Sensei, Graham Sensei, Sakti Sensei, Toda Sensei, Manami Matsuda-san, Timo Abele, Michael He, Vu Huy Hien, Nozomi Jinbo-san, Fajri Koto, Kazuhiro Kobayashi-kun, Nunu Fithria Lubis, Patrick Lumbantobing, Sho Matsumiya, Masahiro Mizukami-kun, Isao Niu, Hiroaki Shimizu, Matthias Sperber, Kou Tanaka, Hoa Vu Trong, Riki Yoshida, and other members of the **Nakamura Lab** at Nara Advanced Institute of Science and Technology (NAIST), as well as Kevin Duh and Jeehoon Kim for the unforgettable summer in Japan.

• Shawn & Kimberely Cheng, Shinwook & Yoojin Kim, Jungho & Joonkyung Yoon and other **friends in New York City** for being my family away from home.

• Sanjoy Dasgupta, my **undergraduate research advisor**, for the first lessons on machine learning and research. I would not be where I am now without him.

The work presented in this thesis has been supported in part by NSF IIS-1111176, NSF IIS-0968450 and NSF IIS-1314778.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In life, we often have to decide what to do, where the issue at hand can be as momentous as settling on a course of actions to be taken for environmental sustainability or as (relatively) insignificant as deciding whether to buy an iPhone or a Samsung Galaxy. When faced with multiple options, we reach conclusions from the knowledge available to us. The process of arriving at conclusions from relevant knowledge, or premises, is called *argumentation*. Throughout the thesis, the term argumentation is mostly used in the monological sense as just described, instead of the dialogical sense denoting the effort by multiple agents to reach a consensus.

Argument is the primary outcome of argumentation. Besnard and Hunter [7] define it as "a set of assumptions (i.e. information from which conclusions can be drawn), together with a conclusion that can be obtained by one or more reasoning steps (i.e. steps of deduction)." Note the use of the word "assumptions" to refer to the information from which the conclusions are derived. These assumptions, so-called premises, are indeed *assumptions*. In other words, premises are not guaranteed to be valid. Thus, they need to be properly evaluated before correct conclusions can be inferred from them, and verifying the correctness of the premises is a crucial step in evaluating an argument.

Evaluating arguments has become an indispensable part of modern life. Our sources of information are no longer limited to books and articles produced by field experts and professional writers—with the advancement of information technology, the amount of textual content written by inexperienced writers, in the form of comments, product reviews, blog posts, etc., is steeply rising. User

generated text is a valuable medium of communication that enables people of various backgrounds to share their experiences and opinions. However, it often consists of unclear argumentative structures and unsubstantiated conclusions, which renders it difficult for us to understand, let alone evaluate. Without properly evaluating the arguments, we cannot determine our stances toward them nor can we build our own arguments based on them.

Unfortunately, the unwieldy amount of user generated text and unclear structures of arguments in it thwart the process of evaluation. This is especially problematic in *eRulemaking*, where citizens comment about rules proposed by government agencies, which are then required to respond to major concerns and criticisms, adjusting the final rule as necessary [47]. The claims in the comments are seriously considered by the audience, government officials and other citizens, only if they are properly supported in a clear manner. In contrast, such a rigorous evaluation of arguments is not as important in many other domains. It is because the decision to be made is not as complex—e.g. deciding which product to buy vs. amending a rule in light of the concerns raised—nor consequential—e.g. an individual or household vs. millions of lives. Because of this, people can rely on statistical summaries of people's arguments. For instance, when you want to see a movie, it is easier, and arguably more effective, to see the average ratings of the movies, instead of reading a select few comments and evaluating the soundness of the arguments.

This thesis proposes an approach to deal with ill-structured arguments in user comments in eRulemaking by automatically extracting and evaluating argumentative structures. This in turn would allow assistance at both ends of communication—guiding the commenters to construct well-structured argu-

[There should be a full ban of peanut products on all airlines,]$_1$ [because peanut allergy could have terrible effects.]$_2$ [Peanut reactions can be life threatening.]$_3$ [Restricting to certain flights is not enough,]$_4$ [as residue from previous flights can remain on the seats.]$_5$ [Recently we flew across the country]$_6$ [and I find left over peanuts in our seats!]$_7$

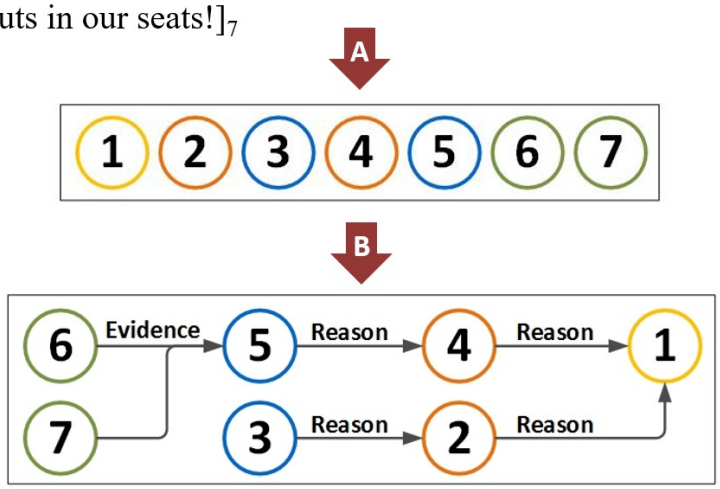

Figure 1.1: Overview of Argument Mining Process

The colored circles represent propositions that suggest courses of action to be taken (yellow), present opinions or value judgments (orange), state factual information (blue), or report personal testimony (green).

ments through real-time feedback and reducing the burden on the readers by means of summarization.

Figure 1.1 shows how a comment would be processed in two stages: (A) proposition classification and (B) support relation identification. The resulting graph representation allows the argumentative structure to be evaluated via a quick comparison to the well-formed argument (See Chapter 2). For instance, Proposition 3 does not have an incoming edge, though it is a factual proposition (denoted by blue) that requires factual evidence. In contrast, Propositions 1, 2, 4, and 5 are properly supported. Propositions 6 and 7 are testimony (denoted

by green), which need not be supported according to the model.

1.1 Contributions

Because the goal of evaluating argumentative structures in online user comments is novel, we aimed to make progress in both the theory and application. Thus, the contributions of this work are threefold: development of a theoretical argumentation model, implementation of computational tools, and construction of publicly available corpora.

Argumentation Model. We propose a novel model of argumentation to capture and evaluate prevalent support structures in user generated text. We first provide a summary of existing argumentation models that are relevant, yet not suitable for our purpose, as part of the literature survey (Chapter 1.2). Then, our model is described in detail (Chapter 2).

In argumentation theory, evaluation typically takes place in the presence of multiple arguments, with the focus on modeling attacking patterns among conflicting arguments [21, 68]. In contrast, our model captures support relations among components of arguments, i.e. various types of propositions serving the function of premise and conclusion, and defines what it means for a practical argument to be well-structured.

Computational Tools. We present computational tools to extract components of our argumentation model from user generated text over several chapters: proposition type classification for determining appropriate types of support for each proposition (Chapter 4), support relation identification for mining support relations among the propositions (Chapter 5), and discourse analysis covering

a broader range of relations among propositions (Chapter 6).

The goal of argumentation mining has been collecting information, just as in similarly named fields, such as data mining and opinion mining. Thus, early argumentation mining systems process news articles, parliamentary records and legal documents, where the documents contain well-formed arguments, i.e. propositions with supporting premises [51, 56, 92, 25, 1]. Even though user generated text as a domain for argumentation mining has become more popular, the goal remains the same: information gathering. Thus only well-formed arguments are targeted. However, our approach enables the identification of any argument, even claims without any explicit premises, and provides ways to evaluate the structure.

Corpora. We constructed several corpora to test the efficacy of our model with real world data (Chapter 3). The corpora comprise the only large scale dataset with user comments from the eRulemaking domain with argumentation-based annotations.

1.2 Background

In this section, we present a survey of related work in argumentation theory (Section 1.2.1), argumentation mining (Section 1.2.2), and eRulemaking, our application domain (Section 1.2.3).

1.2.1 Argumentation Theory

There are several models of argumentation that can be considered for modeling argumentative structures in user generated text. However, they fall short of capturing the prevalent support patterns in practical argumentation and do not allow evaluation of the associated argumentative structures.

Structural Argumentation Models Argumentation modeling is an active area of research that typically focuses on capturing the interaction of arguments via attack and other relations [21, 4]. Among those, structural argumentation models define practical models that can be applied to real text. Besnard and Hunter [6] defines an argument as a pair $< \Phi, \alpha >$ where the set of formulae Φ is the support and α is the consequent of the argument. Such distinction of the premises from the conclusion has become quite a standard over the years. But for the purpose of measuring the evaluability of comments and providing helpful feedback, the interaction of multiple types of elementary units of argument via various support relations is desirable. Another popular model by Prakken [68] differentiates strict from defeasible inference and defines three different attacks on an argument: rebut, undercut, and undermine. However, we are currently focusing on support relations for a single argument for the purpose of constructing more evaluable comments.

The Toulmin Model Several researchers have proposed models of the internal structure of arguments, including Toulmin [78], Farley and Freeman [26], and Reed and Walton [83]. One of the most widely known argumentation models is the Toulmin Model [78].

It models an argument as an interplay of 6 elements:

1. *Claim (C):* a proposition being argued for.
2. *Data (D):* objective evidence in support of a claim. The objective evidence can come in various forms, including experiment data and personal testimony.
3. *Warrant (W):* a justification for inferring the claim from the data.
4. *Backing (B):* objective evidence in support of a warrant.
5. *Qualifier (Q):* words or phrase showing the confidence level of the arguer with respect to the claim being made.
6. *Rebuttal (R):* conditions under which the claim may not be true.

The first four elements are claimed to exist in any argument, either explicitly or implicitly, whereas *Qualifier* and *Rebuttal* are optional [79].

As this model has been receiving much attention, many extensions have been proposed. For instance, Bench-Capon [3] added an additional component called "presupposition component" denoting a necessary assumption for the argument that is to be taken without dispute, and Freeman [26] identified subcategories of *Warrant* to distinguish various types of warrants. One major issue with the Toulmin Model is that it is underspecified in a few ways, and this is problematic for implementation. Even the experts cannot agree on the correct interpretation, especially about the *Warrant*. For instance, Hitchcock [30] considers it an inference-license, not a premise, whereas Eemeren et al. [80] claim that *Warrant* is indistinguishable from *Data*. In our model, we clearly define the elementary units and their interactions.

Argumentation Schemes Argumentation schemes provide templates for

prominent patterns of arguments, defining specific premises and a set of critical questions for each scheme [83, 9, 85].

For example, *Argument from Expert Opinion* consists of 3 premises in support of a conclusion, as well as 6 critical questions [84]:

1. Scheme
    - Premise : E is an expert in D.
    - Premise : E asserts that A is known to be true.
    - Premise : A is within D.
    - Conclusion : Therefore, A may plausibly be taken as true.
2. Critical Questions
    - Expertise: How credible is E as an expert source?
    - Field: Is E an expert in the field that A is in?
    - Opinion: What did E assert that implies A?
    - Trustworthiness: Is E personally reliable as a source?
    - Consistency: Is A consistent with what other experts assert?
    - Backup Evidence: Is E's assertion based on evidence?

The critical questions make argumentation schemes useful for assessing the validity or strength of arguments, and can provide a more detailed assistance to commenters. However, given comments consisting of arguments with only a few or no premises explicitly stated, it is practically impossible to decide which argumentation scheme matches the commenters' intentions, and this in turn

means that we cannot easily identify relevant critical questions for given comments. Thus, we need a new argumentation model for the purpose of evaluating the internal structures of practical arguments.

1.2.2 Argumentation Mining

Argument Mining aims at developing methods and techniques for automatic extraction of arguments from texts in natural language. An argument is a complex discourse unit with boundaries easily recognisable by humans and yet hard to determine by a computer. For this reason, argument mining is often supported with rhetorical document structure, argument schemes or dialogical relations.

This area of research began to attract attention over a decade ago. Argumentative zoning [76, 77] looked at recognising argumentative discourse units from unstructured scientific papers using rhetorical structure of a document. The results varied from the highest, F-score of 0.86 for the recognition of parts of papers in which an author refers to their own research to as low as F-score of 0.26 for the recognition of parts in which an author presents arguments against other approaches. The authors point out that their solution is domain-specific and works well for academic papers, as it relies on specifically tailored sentential features.

Automated classification of sentences as either argument or non-argument [52] on the material from discussion fora, legal judgements, newspapers, parliamentary records and weekly magazines achieved an average accuracy of 70% using maximum entropy and multinomial naive Bayes classifiers. In this study,

the score for discussion fora (68.4%) was lower than for the newspaper articles (73.22%). The authors suggest that discussion fora contain more ambiguous arguments and are less well-formed texts compared to the news and legal texts. Classification of sentences as argument or non-argument constitutes the first step in argument mining, however it does not yet provide the information about argumentative relations, such as reason-conclusion structure or conflict.

The relations between reason and conclusion in legal texts are explored in [57]. The first step in the argument detection task was the usage of Naive Bayes as a statistical classifier, achieving 73% accuracy on the Araucaria corpus and 80% on the ECHR corpus (a corpus of texts issued by the European Court of Human Rights). The next step was to classify extracted propositions by their argumentative function. Automatic marking of clauses as either premises or conclusions reached close to 70% accuracy. In the third step, to determine the global argumentative structure a set of manually crafted rules was used, which created a context-free grammar. In this task 60% accuracy was obtained.

Since 2014, the area of argument mining has been witnessing a rapidly increasing interest. Analysis of support and attack relations in the corpus of German argumentative microtexts [63] provided a highest achieved F-score of 0.7. Automated extraction of counter-consideration is explored in [64]. A speaker may provide counter-consideration to her own statement in anticipation of the critique. This study provides evidence that lexical indicators (especially "but", also "however" and "although") perform well as predictors of counter-considerations.

A new field of Argument Mining is mining arguments from dialogue [12] which explores how argumentative structures are built upon dialogical struc-

tures in interaction. Previous analysis of the structure of dialogues (for example of ad hominem dialogues [13]) led to the description of argument structure in dialogue protocol. This allowed for the theoretical foundations for argument mining in dialogue as well as its initial implementations described in [11]. Argument mining from dialogue is of particular importance for the automated extraction of conflict and controversy, as we believe that those can only emerge through interaction and are inherently related to dialogical structures.

1.2.3 eRulemaking

While user generated text appears in many platforms, from tweets to product reviews, we focus on user comments from a particular domain, eRulemaking. The reason is that presenting well structured arguments with adequately supported claims is especially important for the targeted audience of the domain.

Rulemaking is a multi-step process that federal agencies use to develop new regulations on health and safety, finance, and other complex topics. From its inception, rulemaking was designed to be a participatory process for making policy, as opposed to a purely bureaucratic one [24]. To ensure public awareness of proposed regulations, federal agencies are required to publish materials describing the legal basis, factual and technical support, policy rationale, and costs and benefits of a proposal. Agencies must specify a comment period, usually 60 to 90 days, during which anyone may send the agency comments. Further, agencies are required to respond to information, arguments, and criticisms presented by the public as part of its final rule [47].

This process gives citizens notice of proposed regulations that could affect

them, and provides them with an opportunity to meaningfully influence the content of that regulation. However, the participants most involved in the rulemaking process have traditionally been sophisticated citizens — those well-resourced members of industry and large advocacy or lobbying organizations who know how the process works and understand how to present their data and arguments persuasively in a comment to the agency. These citizens have the organizational capacity, economic or political interest, and proximity to policymakers to learn about upcoming rulemakings and respond point-by-point to an agency's proposal.

The comparatively low engagement of non-expert (or, at least, less well-resourced) citizens—individuals, including small business owners, state and local government entities, and non-governmental organizations—has presented problems for the participatory nature of rulemaking [62]. Moreover, the notice-and-comment rulemaking process cannot fully meet the goal of developing regulations that are as tailored and effective as possible when many interested citizens do not know about the rulemaking process, much less how to effectively engage in it.

eRulemaking leverages information technology to increase public awareness of and participation in federal rulemaking—a multi-step process that federal agencies use to develop new rules, incorporating the feedback from citizens directly affected by the proposed rules [47]. Immediate access to materials about a proposed rule, as well as the ability to share them widely and instantaneously, should increase awareness and participation among citizens who have been missing from the off-line process. One would also expect that the flexibility of time to read, reflect on, and respond to an agency proposal should simulta-

neously increase the quality of that participation.

Yet, experience demonstrates that merely putting proposed rules and their supplemental materials online has not been enough to overcome the barriers that non-expert citizens face when trying to participate in what is often a highly technocratic process [17]. Without knowing the expectations for participating in rulemaking, non-experts often default to "voting and venting" behaviors—expressing their outcome preferences or identifying problems but not providing additional data, information, arguments, or reasons that could substantiate their positions [24]. Because rulemaking is a reasoned decision-making process, and agencies are required to weigh reasoning and evidence, arguments that do not explicitly state reasons or neglect to provide objective evidence for factual claims are not influential. Such arguments prevent effective communication with other participants, as well.

One approach to make the comments more suitable is to introduce human moderation: Cornell eRulemaking Initiative (CeRI) partners with federal agencies to host online discussions of ongoing rulemakings on its civic engagement platform, *regulationroom.org*, with active moderators interacting with the commenters. A key role of the moderators is to prompt commenters to better support the propositions they make, asking for either a reason or evidence. Though human moderation can be effective, hiring and training human moderators can be cost intensive. Also, quicker moderation is desirable: A majority of the commenters are one-time visitors who never return to the website[1], thus, the moderation that takes place after the commenters leave can be ineffective.

As an alternative, automated extraction of arguments is proposed in this

[1]Of the 12,665 total visits to *regulationroom.org* to discuss a proposed *Home Mortgage Consumer Protection* rule, 8,908 corresponded to unique visitors.

thesis. In the subsequent chapters, we first define the theoretical model of argumentation to capture prevalent support structures in user generated text (Chapter 2). This model defines the structure of a well-structured argument, while being flexible enough to capture practical arguments consisting of claims with a subset or all of their premises missing. After a survey of corpora created for the experiments (Chapter 3), software implementations to automatically extract the components of the model are presented: proposition type classification for determining appropriate types of support for each proposition (Chapter 4), support relation identification for mining support relations (Chapter 5), and discourse analysis covering a broader range of relations among propositions (Chapter 6).

CHAPTER 2

ARGUMENTATION MODEL OF EVALUABILITY

eRulemaking is an ongoing effort to use online tools to foster broader and better public participation in rulemaking — the multi-step process that federal agencies use to develop new health, safety, and economic regulations. The increasing participation of non-expert citizens, however, has led to a growth in the amount of arguments whose validity or strength are difficult to evaluate, both by the government agencies and fellow citizens. Such arguments typically neglect to provide reasons for their conclusions and objective evidence for the factual claims upon which the arguments are based. In this chapter, we propose a novel argumentation model for capturing the *evaluability*—the ability to be evaluated—of user comments in eRulemaking. More specifically, an argument is *evaluable* if its conclusion is supported with at least one explicit premise of an appropriate type defined in the model.[1] This model is intended to be used for implementing automated systems to assist users in constructing evaluable arguments in an online commenting environment for the benefit of quick feedback at a low cost.

2.1 Background

eRulemaking leverages information technology to increase public awareness of and participation in federal rulemaking—a multi-step process that federal agencies use to develop new rules, incorporating the feedback from citizens directly affected by the proposed rules [47]. Immediate access to materials about a pro-

[1] The underlying assumption is that if at least one premise of the right type is explicitly stated, the readers can see the writer's general approach to support their claim. Requiring further support would be ideal, but too rigorous for practical argumentation.

posed rule, as well as the ability to share them widely and instantaneously, should increase awareness and participation among citizens who have been missing from the off-line process. One would also expect that the flexibility of time to read, reflect on, and respond to an agency proposal should simultaneously increase the quality of that participation.

Yet, experience demonstrates that merely putting proposed rules and their supplemental materials online has not been enough to overcome the barriers that non-expert citizens face when trying to participate in what is often a highly technocratic process [17]. Without knowing the expectations for participating in rulemaking, non-experts often default to "voting and venting" behaviors—expressing their outcome preferences or identifying problems but not providing additional data, information, arguments, or reasons that could substantiate their positions [24]. Because rulemaking is a reasoned decision-making process, and agencies are required to weigh reasoning and evidence, arguments that do not explicitly state reasons or neglect to provide objective evidence for factual claims are not influential. Such arguments prevent an effective communication with other participants, as well.

To better understand the problem, let's consider short snippets of user comments about an Airline Passenger Rights rule by Department of Transportation collected from an eRulemaking platform, *regulationroom.org*:

> (1) All airfare costs should include the passenger's right to check at least one standard piece of baggage.$_A$ All fees should be fully disclosed at the time of airfare purchase, regardless of nature (i.e. optional or mandatory).$_B$ Any changes in fees should be identified by air carriers at least 6 months prior to taking effect.$_C$

Because this comment consists purely of claims without any support, it is difficult to evaluate its strength, making it neither influential nor useful for the lawmakers. (In argumentation terminology, there are three seemingly independent arguments, each consisting of a conclusion without any explicit premises.) This is unfortunate, as the commenter already took the time and effort to participate in eRulemaking process, yet hardly any benefit was produced. Had the commenter made the supporting premises explicit, the arguments would have been better assessed and more valuable for the lawmakers.

> (2) I would support a full ban of peanut products on any airline.$_A$ Peanut reactions can be life threatening.$_B$ An individual doesn't have to consume the product to have a life threatening reaction.$_C$ They can have contact or inhalation reactions.$_D$ Restricting to certain flights is not enough to protect the passengers,$_E$ as residue can be rampant.$_F$

This comment is much more evaluable, as the premises for the conclusion to fully ban peanut products on airlines are clearly stated. (There are conclusions from sub-arguments, as well, but we will discuss them in more detail when we revisit this example.) To fully assess the argument, however, the readers will need to verify the factual claims such as 2.F, and perhaps 2.B, depending on the reader's background knowledge. Thus, providing evidence, such as a URL or a citation of an accredited source, for those claims would have made the evaluation process easier.

> (3) There should definitely be a cap and not this hideous amount between $800 and $1200.$_A$ $400 is enough compensation,$_B$ as it can cover a one-way fare across the US.$_C$ I checked in a passenger on a $98.00 fare from east coast to Las Vegas the other day.$_D$

The is a clearly written comment that can be adequately evaluated as it is. One thing that can be added is, perhaps, evidence for 3.D.

One approach for making the comments more suitable for assessment is to introduce human moderation: Cornell eRulemaking Initiative (CeRI) partners with federal agencies to host online discussions of ongoing rulemakings on its civic engagement platform, *regulationroom.org*, with active moderators interacting with the commenters. A key role of the moderators is to prompt commenters to better support the proposition they make, asking for either a reason or evidence. Though human moderation can be effective, hiring and training human moderators can be cost intensive. Also, a quicker moderation is desirable: A majority of the commenters are one-time visitors who never return to the website[2], and thus, the moderation that takes place after the commenters leave can be ineffective.

In this section, we propose an argumentation model capturing the evaluability of arguments. This model is intended to be used for implementing automated systems to assist users in constructing evaluable arguments under online commenting environment for the benefit of quick feedback at a low cost.

As discussed in Section 1.2.1, some of the existing argumentation models are relevant for this purpose, yet none is sufficient. To summarize the issues, argumentation models in general model conflicts among multiple arguments. Even though structural argumentation models model the inner structure of each argument, they are too simple, e.g. the only elementary unit types are premise and claim, to capture the evaluability of the argumentative structures. While the Toulmin Model contains a more diverse set of elementary units [78, 79], the ele-

[2]Of the 12,665 total visits to *regulationroom.org* to discuss a proposed *Home Mortgage Consumer Protection* rule, 8,908 corresponded to unique visitors.

mentary units cannot be clearly distinguished from one another [80, 30]. Lastly, argumentation schemes do define detailed argumentative structures. However, identifying the scheme in use is difficult due to many implicit premises comprising practical arguments, especially in user comments.

We now present an argumentation model capturing the evaluability of arguments in user comments with various elementary units and support relations.

2.2 Elementary Units

We adopt and modify results from argumentation research that classifies different types of claims in order to study their characteristics [32, 91]. Hollihan and Baaske, for instance, distinguish three types of claims: fact, value and policy. Simply put, fact claims are verifiable with objective evidence, value claims express preference, interpretation or judgment, and policy claims assert a course of action to be taken. (As we describe each type of elementary units, please refer to examples from Table 2.1.)

For our purpose, however, we distinguish fact claims about personal state or experience and non-experiential ones and accept the former as a form of evidence and thus not require any further support. The reasons are threefold: (1) it is often practically impossible for the commenters to provide evidence for fact claims about personal state or experience. One reason is that people normally do not have evidence for what they experience (See TESTIMONY 3, for example). This is especially true if we restrict the eRulemaking interface to a typical online commenting environment, where only textual inputs are accepted. Thus, even if one had a picture of leftover peanuts on their seat, they cannot upload

Unit	Support	#	Example
POLICY	REASON	1	Peanuts should be banned from all airlines.
		2	Do not force passengers to risk their health.
		3	Government needs to protect their citizens.
VALUE		1	Global warming is more important than any other pressing issues we are facing.
		2	They will lose business eventually.
		3	I am not happy with my new pet.
FACT	EVIDENCE	1	Food allergies are seen in less than 20% of the population
		2	The report states that peanut can cause severe reactions.
		3	The governor said that the economy will recover soon.
TESTIMONY	EVIDENCE*	1	I've been a physician for 20 years.
		2	My son has hypolycemia.
		3	There were leftover peanuts from the previous flight on my seat.
REFERENCE	NONE	1	http://www.someurl.com/somewebpage.html
		2	J. Doe 2014
		3	J. Doe 2014. Paper Title. In Proceedings of Conference Name. Pages 12-25

Table 2.1: Appropriate Support Type and Examples of Each Elementary Unit Type

*Optional Evidence

it as evidence for TESTIMONY 3. Another reason is that a sufficient evidence may violate privacy, as in the case of TESTIMONY 1 and 2. (2) In eRulemaking, lawmakers accept a wide variety of comments from citizens, including accounts of personal experience relevant for proposed rules. Arguments based on such *anecdotal evidence* are exactly the type of information that is valuable, yet cannot be obtained through the lawmakers' usual channel of communication with domain experts. If these accounts are relevant and plausible, the agencies may use them, even if they are not substantiated with evidence. (3) Toulmin and Hitchcock classifies them as justified grounds, as well (See Table 2.2).

Note that, because a policy claim expresses a specific type of judgment—

Justified Grounds from [31]	Type	Justified Grounds from [79]	Type
Direct observation	*	Experimental observations	REFERENCE
Written records of direct observation	REFERENCE	Matters of common knowledge	FACT***
Memory of what one has previous observed	*	Statistical data	REFERENCE
Personal testimony	TESTIMONY	Personal testimony	TESTIMONY
Previous good reasoning or argument	Any	Previously established claims	Any
Expert opinion	**	Other comparable "factual data"	REFERENCE
Appeal to an authoritative reference source	REFERENCE		

Table 2.2: Elementary Unit Types of Justified Grounds (Evidence)

* This cannot be part of an argument. The moment you state your observation, it becomes a testimony, not a memory. Thus, testimony can be part of an argument, but memory cannot be.

** If there is a written record, which should be the case for established expert opinions, it is REFERENCE. If a local expert expressed an opinion to the arguer, or he is an expert himself, it is TESTIMONY.

*** As there is no knowledge base of common knowledge, factual propositions about a common knowledge cannot be distinguished from the rest. Thus, FACT is not considered as evidence in the automated system.

the one that asserts what should be done—it can be considered a type of value claim. Then, we have a good match between the claim types and appropriate support relations: Since fact claims are verifiable, the best form of support is evidence, in the form of a reference to an accredited source, showing the claim is truthful. On the other hand, no such evidence exist for value and policy claims as they are unverifiable by definition. Thus, an appropriate support is a reason

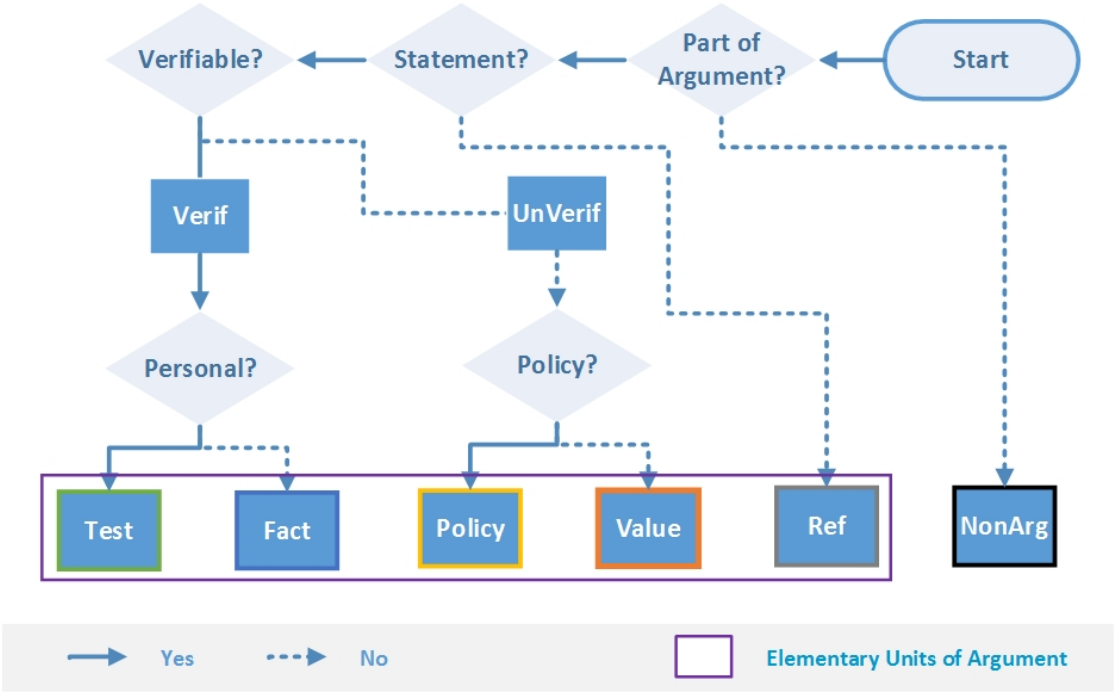

Figure 2.1: Elementary Units

from which the claim can be inferred[3].

Lastly, we add a type called REFERENCE to encompass URLs and citations of published articles, as most factual evidence in online comments is provided in this form. REFERENCE and TESTIMONY are the only elementary units that qualify as evidence. And this completes the set of five elementary units for our model as follows: (A flowchart for deriving of elementary units is shown in Figure 2.1.)

Proposition of Non-Experiential Fact (FACT) : A proposition of fact is an objective proposition where *objective* means "expressing or dealing with facts or

[3]Even though the appropriate support type for both policy and value claims is reason, their distinction is retained for a possible extension of the system: The system can guide commenters to explicitly suggest a course of action, instead of simply making a value judgment on the proposed rules. However, this may not be necessary, as value claims made about different aspects of proposed rules typically make it obvious what course of action the commenter prefers.

conditions as perceived without distortion by personal feelings, prejudices, or interpretations."[4] By definition a FACT has truth values that can be verified with objective evidence. We restrict the notion of verifiability to the evidence potentially being available at the present time. Thus, predictions about the future are considered unverifiable. The examples show various types of propositions that can be proved with direct objective evidence[5]. Note that, FACT 3 is considered a FACT because whether or not the governor *said*, "The economy will recover soon." can be verified with objective evidence, even though his speech itself contains a value judgment.

Proposition of Experiential Fact (TESTIMONY[6]) : Objective proposition about the author's personal state or experience. One major characteristic of this type of objective propositions, as opposed to the non-experiential ones classified as FACT, is that it is often practically impossible to provide objective evidence proving them: It is unrealistic to expect an objective evidence for a personal experience to exist in the public domain, and thus, one often does not have the evidence. For instance, you would not expect there to be any evidence for TESTIMONY 3. Also, the author may not want to reveal the evidence for privacy reasons (See TESTIMONY 1 and 2).

Proposition of Value (VALUE) : Proposition containing value judgements without making specific claims about what should be done (If so, then it is a POLICY.). Because of the subjectivity of value judgements, a VALUE cannot be proved directly with objective evidence; however, providing a reason as support is feasible and appropriate. Consider VALUE 1, for instance. There is no

[4]http://www.merriam-webster.com/

[5]See Section 2.3 for what constitute *objective evidence*.

[6]Technically a better term would be OBJECTIVE TESTIMONY, but we use TESTIMONY for the ease of use.

objective evidence that can directly prove the proposition, because even if you were to provide objective evidence showing negative effects of global warming, subjective judgment must be made to reach the conclusion that it is the most important issue. VALUE 2 is considered unverifiable, because as discussed in the FACT paragraph, objective evidence need to be able to exist at the present time. For VALUE 2 the objective evidence will be available only in the future. An expression of private state, such as VALUE 3, is similar to propositions of value in this respect, thus are categorized as VALUE includes opinions as well as proposition of value[7].

Proposition of Policy (POLICY) : Assertion that a specific course of action should be taken. It almost always contains modal verbs like "should" and "ought to." Just like VALUE, a POLICY cannot be directly proved with objective evidence, and a proper type of support is a logical reason from which the proposition can be inferred. You can present objective evidence about a similar event that has taken place to make an analogy, but it is still not a direct proof that the same thing will happen again. In other words, the existence of a similar event can only be an indirect evidence for the assertion insufficient on its own, not a direct proof for it.

Reference to a Resource (REFERENCE) : reference to a source of objective evidence. In online comments, a REFERENCE is typically a citation of a published work or a URL for online documents. Quotes or paraphrase of a reference such as FACT 2 or 3 are not REFERENCE, as whether the given resource contains the claimed content is a factual statement that can be verified. REFERENCE could also be an attachment if the commenting interface allows it. As it is shown in

[7]The motivation for the classification of propositions is to determine the desirable types of support: If the desirable types of support are the same, they should be classified into the same category.

Table 2.2, REFERENCE is the elementary unit category for the most types of justified grounds.

2.3 Support Relations

As discussed in the previous section, the elementary units are distinguished with the following types of support in mind.

Reason: An elementary unit X is a *reason* for proposition Y if Y explains why X is true. For example, FACT 2 can be a reason for POLICY 1. To show a FACT proposition is true, the strongest form of support is objective evidence showing that the claim is true, not a reason explaining why the conclusion is true, as such inferences in practical reasoning are often defeasible.

Evidence: X, a set of elementary units of type TESTIMONY or REFERENCE, is *evidence* for a proposition Y if it confirms that proposition Y is valid or not. For example, evidence for FACT 1 can be a citation or link to a medical research showing the percentage of the population with food allergies is less than 20%. The possible types of evidence are limited to TESTIMONY or REFERENCE based on previous studies on what constitute justified grounds [79, 31]. See Table 2.2 for how the list of justified grounds map to our classification of elementary units of argument.

2.4 Formalization

Let *Proposition* = {POLICY, VALUE, FACT, TESTIMONY}, *Evidence* = {TESTIMONY, REFERENCE}, and *Type()* a function that maps argumentative propositions[8] to the set of elementary units.

Definition 2.4.1 *An **argument** is a set {R, E, c} where:*

1. *c is the conclusion such that Type(c) ∈ Proposition.*
2. *R is a set of reasons explaining that c is true, such that* $\forall r \in R,\ Type(r) \in Proposition$.
3. *E is a set of evidence confirming that c is true, such that* $\forall e \in E,\ Type(e) \in Evidence$.

Definition 2.4.2 *Let A={R, E, c} be an argument. The set of **sub-arguments** of A is defined recursively as the union of* $\{R'_i,\ E'_i,\ r_i\}$ *for* $\forall r_i \in R$, *such that* $Type(r_i) \in Proposition$, *and each of their sub-arguments.*

Definition 2.4.3 *An **evaluable argument** A is an argument {R, E, c} where at least one of the following is true for A and all its sub-arguments:*

1. *Type(c) =* TESTIMONY
2. *Type(c) ∈* {POLICY, VALUE}, *and* $R \neq \emptyset$, *such that* $\forall r \in R,\ Type(r) \in Proposition$
3. *Type(c) =* FACT, *and* $R \neq \emptyset$, *such that* $\forall r \in R,\ Type(r) \in$ FACT
4. *Type(c) =* FACT, *and* $E \neq \emptyset$, *such that* $\forall e \in E,\ Type(e) \in Evidence$

[8] An argumentative proposition is a proposition that is part of an argument.

In other words, an argument can consist of zero or more number of reasons and pieces of evidence, but there are a few restrictions that must be met in order for it to be properly assessed. When the conclusion is a TESTIMONY, explicit premises need not be provided in order for the argument to be assessed. (As discussed, we take TESTIMONY as a type of objective evidence.) Conclusions of all other types need at least one type of support: POLICY and VALUE require an explicit premise as support, and FACT can be supported with evidence or another FACT. (See Example 2.C and D.) The underlying assumption is that if each proposition has at least one supporting reason or evidence, understanding the argument and assessing it is much more feasible than the case in which no explicit premise or evidence is given.

Table 2.3 shows how the comment examples from Section 2.1 are processed according to the argumentation model. The last column lists what additional support is needed to make the argument evaluable as defined. All three conclusions in Comment 1 need support in the form of reason, and providing evidence or reason for three conclusions from Comment 2 will make the argument more evaluable. Comment 3 is a well written comment that can benefit from adding evidence for Proposition D, but it is only optional.

2.5 Conclusions

eRulemaking is an ongoing effort to use online tools to foster broader and better public participation in rulemaking. The increasing participation of non-expert citizens, however, has led to a growth in the amount of arguments whose validity or strength are difficult to evaluate, both by the government agencies and

c	Type	Appropriate Support	Existing Support		Needed Support
			P	E	
Example Comment 1					
A	POLICY	P	∅	∅	P*
B	POLICY	P	∅	∅	P*
C	POLICY	P	∅	∅	P*
Example Comment 2					
A	POLICY	P	{B,C,E}	∅	
B	FACT	E or P	∅	∅	E or P**
C	FACT	E or P	{D}	∅	
D	FACT	E or P	∅	∅	E or P**
E	FACT	E or P	{F}	∅	
F	FACT	E or P	∅	∅	E or P**
Example Comment 3					
A	POLICY	P	{B}	∅	
B	FACT	E or P	{C}	∅	
C	FACT	E or P	∅	{D}	
D	TESTIMONY	(E)***	∅	∅	(E)***

Table 2.3: Comment Examples Processed According to the Argumentation Model

* POLICY, VALUE, FACT or TESTIMONY

** If E: TESTIMONY or REFERENCE, If P: FACT

*** Optional

fellow citizens. To support the implementation of automated systems to assist users in constructing evaluable arguments under online commenting environment for the benefit of quick feedback at a low cost, we propose an argumentation model capturing the evaluability of arguments. For future extensions of our system, considering the potential attacks to help users construct arguments that are harder to defeat can be interesting.

CHAPTER 3

CORPORA

Since we tackle novel tasks based on a new model of argumentation we propose, a significant effort was put into building various corpora. All of the experiments were done with these corpora, except for the results presented in Chapter 6 in which we use an existing dataset for a fair comparison with existing literature. The corpora are summarized in Table 3.1, and each corpus is described in detail in the following sections.

Corpus	Cornell eRulemaking Corpus - APR & MS
Rule	Airline Passenger Rights, Mortgage Services
Annotation	Proposition Types : Unverifiable, Verifiable-Experiential, and Verifiable-Nonexperiential
Publications	Park and Cardie [60], Park et al. [61]
URL	http://www.joonsuk.org
Corpus	Cornell eRulemaking Corpus - CDCP
Rule	Consumer Debt Collection Practices
Annotation	Proposition Types : Policy, Value, Fact, Testimony, and Reference
	Intra-comment Support Relations
Publications	Under Preparation
URL	N/A
Corpus	eRulemaking Controversy Corpus
Rule	Airline Passenger Rights
Annotation	Intra-comment Support Relations, Inter-comment Conflict Relations
Publications	Konat et al. [39], Under Review
URL	http://arg.tech/ercctrain http://arg.tech/ercctest

Table 3.1: Summary of Corpora

3.1 Cornell eRulemaking Corpus - APR & MS

For Step A in Figure 1.1, we have collected and manually annotated statements from user comments extracted from an eRulemaking website, *Regulation Room*[1].

Regulation Room is an experimental website operated by Cornell eRulemaking Initiative (CeRI)[2] to promote public participation in the rulemaking process, help users write more informative comments and build collective knowledge via active discussions guided by human moderators. *Regulation Room* hosts actual regulations from government agencies, such as the U.S. Department of Transportation.

For our research, we collected and manually annotated 9,476 statements from 1,047 user comments from two recent rules: Airline Passenger Rights (tarmac delay contingency plan, oversales of tickets, baggage fees and other airline traveller rights) and Home Mortgage Consumer Protection (loss mitigation, accounting error resolution, etc.).

To start, we collected 1,147 comments and randomly selected 100 of them to devise an annotation scheme for identifying appropriate types of support for statements and to train annotators. Initially, we allowed the annotators to define the span for a "statement", leading to various complications and a low inter-annotator reliability. Thus, we introduced an additional step in which comments were manually sliced into "statements" before being given to the annotators. "Statements" were determined using line breaks and punctuation as guides. A "statement" found this way was split further if it consisted of two or more independent clauses. The sliced comments were then coded by two

[1] http://www.regulationroom.org
[2] http://www.lawschool.cornell.edu/ceri/

annotators into the following four disjoint classes:

	#	Statement
VERIF$_{EXP}$	1	I've been a physician for 20 years.
	2	My son has hypolycemia.
	3	They flew me to NY in February.
	4	The flight attendant yelled at the passengers.
VERIF$_{NON}$	5	*They can have inhalation reactions.*
	6	since *they serve them to the whole plane.*
	7	*Peanuts do not kill people.*
	8	Clearly, *peanuts do not kill people.*
	9	I believe *peanuts do not kill people.*
	10	*The governor said* that he enjoyed it.
	11	*food allergies are rare*
	12	*food allergies are seen in less than 20% of the population*
UNVERIF	13	Again, keep it simple.
	14	Banning peanuts will reduce deaths.
	15	I enjoy having peanuts on the plane.
	16	others are of uncertain significance
	17	banning peanuts is a slippery slope
OTHER	18	Who is in charge of this?
	19	I have two comments
	20	http://www.someurl.com
	21	Thanks for allowing me to comment.
	22	- Mike

Table 3.2: Example Sentences.

* Italics is used to illustrate *core clause*.

Verifiable Statement [Experiential(VERIF$_{EXP}$) and Non-experiential(VERIF$_{NON}$)]. A statement is verifiable if it contains an objective assertion, where *objective* means "expressing or dealing with facts or conditions as perceived without distortion by personal feelings, prejudices, or interpretations."[3] Such assertions have truth values that can be proved or disproved with objective evidence[4]:

Consider the examples from Table 3.2. Statements 1 through 7 are clearly verifiable because they only contain objective assertions. Statements 8 and 9 show that adding subjective expressions such as "Clearly" (e.g. statement 8)

[3]http://www.merriam-webster.com/

[4]The correctness of the assertion or the availability of the objective evidence does not matter.

or "I believe that" (e.g. statement 9) to an objectively verifiable statement (e.g. statement 7) does not affect the verifiability of the statement. Statement 10 is considered verifiable because whether or not the governor *said* "he enjoyed the peanuts" can be verified with objective evidence, even though whether he really did or not cannot be verified.

For the purpose of identifying an appropriate type of support, we employ a rather lenient notion of objectivity: an assertion is objectively verifiable if the *domain of comparison* is free of interpretation. For instance, statement 11 is regarded as objectively verifiable, because it is clear, i.e. it is not open for interpretation, that *percentage of the population* is the metric under comparison even though the *threshold* is purely subjective[5]. The rationale is that this type of statement can be sufficiently substantiated with objective evidence (e.g. published statistics showing the percentage of people suffering from food allergies). Another way to think about it is that statement 11 is a loose way of saying a (more obviously) verifiable statement 12, where the commenter neglected to mention the threshold. This is fundamentally different from statements 13 through 16 for which objective evidence cannot exist[6].

A verifiable statement can further be distinguished as experiential or not, depending on whether the statement is about the writer's personal state or experience (VERIF_{EXP}) or is a more general statement (VERIF_{NON}). This difference determines whether objective evidence is mandatory or optional with respect to the credibility of the comment. Evidence is optional when the evidence contains private information or is practically impossible to be provided: While

[5]One may think anything less frequent than the average is rare and another may have more stricter notion.

[6]Objective evidence may exist for statements that provide *reasons* for statements 13 through 16.

statements 1 through 3 can be proved with pictures of official documents, for instance, the commenters may not want to provide them for privacy reasons. Also, the website interface may not allow pictures to be uploaded in comment section, which is the case with most websites. Statement 4 is practically impossible to prove unless the commenter happened to have recorded the conversation, and the website interface allows multimedia files to be uploaded. This is different from statements 5 through 12, which should be (if valid, that is) based on non-experiential knowledge the commenter acquired through objective evidence available to the public.

In certain domains, VERIF$_{EXP}$ statements—sometimes referred to as *anectotal evidence*—provide the novel knowledge that readers are seeking. In eRulemaking, for instance, agencies accept a wide variety of comments from the public, including accounts of personal experience with the problems or conditions the new regulation proposes to address. If these accounts are relevant and plausible, the agencies may use them, even if they include no independent substantiation. Taking it to an extreme, even if the "experience" is fake, the "experience" and opinions based on them are valuable to the agencies as long as the "experience" is realistic.

Unverifiable Statement (UNVERIF). A statement is unverifiable if it cannot be proved with objective evidence. UNVERIF statements are typically opinions, suggestions, judgements, or assertions about what will happen in the future. (See statements 13 through 17.) Assertions about the future are typically unverifiable, because there is no direct evidence that something will happen. A very prominent exception is a prediction based on a policy of organizations, i.e. "The store will be open this Sunday." where the policy serves as a direct evidence.

Other Statement (OTHER). A statement is in this category if it does not belong to any of the aforementioned categories, i.e. it cannot be verified with objective evidence and no supporting reason is required for the purpose of improving the comment quality. Examples include question, greeting, citation, and URL. (See statements 18 through 21.)

The resulting distribution of classes is shown in Table 3.3. Note that even though we employed a rather lenient definition of objective statements, the distribution is highly skewed towards UNVERIF statements. This is expected because the comments are written by people who want to express their opinions about a regulation. Also, OTHER statements comprise about 7% of the data, suggesting that most comment statements are argumentative. Such a high percentage of argumentative propositions is a result of considering unsupported claims as argumentative. Since our goal is to evaluate argumentative structures, it is important to consider such "ill-structured" arguments as arguments. This is rarely done in argumentation mining, where the goal is to gather claims and their respective support.

The inter-coder reliability checked on 30% of the data is moderate, yielding an *Unweighted Cohen's* κ of 0.73. Most of the disagreement occurred in statements like "Airlines have to provide compensation for both fees and lost bags" in which it is not clear from the context whether it is an opinion (UNVERIF) or a law (VERIF_{NON}). Also, opinions that may be verifiable (e.g. "The problems with passenger experience are not dependant on aircraft size!") seem to cause disagreement among annotators.

Regulation	VERIF$_{NON}$	VERIF$_{EXP}$	UNVERIF	Subtotal	OTHER	Total	# of Comments
APR	1106	851	4413	6370	522	6892	820
HMCP	251	416	1733	2400	186	2586	227
Total	1357	1267	6146	8770	708	9476	1047

Table 3.3: Class Distribution Over Statements

3.2 Cornell eRulemaking Corpus - CDCP

For Step A and B in Figure 1.1, we have annotated comments about Consumer Debt Collection Practices rule. The annotation scheme is described in detail in Chapter 2.

The data consists of user comments (about the consumer rights in receiving financial services) crawled from an eRulemaking website, *http://www.regulationroom.org*, after the commenting period was closed.

(Ex 1) [Knowingly calling third parties should be prohibited across the board.]$_{\text{claim}}$ [It is no one else's business what goes on between creditors and their debtors.]$_{\text{premise}}$

(Ex 2) [[Someone who lived at my address, more than 30 years ago had a debt with Household Finance.]$_1$ [We get letters that we return "Addressee Unknown" and phone calls.]$_2$ [Finally I called one number back and told them that the person they looked for wasn't there.]$_3$ [That law firm stopped robo-calling]$_4$ [but a few months later a new one started up.]$_5$]$_{\text{premise}}$ [Records should convey when debts are sold from collection agency to collection agency.]$_{\text{claim}}$

The annotators identified and annotated proposition boundaries as well as

support relations between the propositions. Here, a proposition may be a full sentence, like the premise in (Ex 1), or a clause, like proposition 5 in (Ex 2). The annotation was conducted with the GATE[7] annotation tool [19].

Each comment was annotated by two annotators, then a third annotator manually resolved the conflicts to produce the final set of annotations. The inter-annotator agreement was measured by taking the average of the accuracy with respect to annotator A's annotations (i.e. percentage of annotator A's annotations that *matches* an annotation by annotator B) and that with respect to annotator B's annotations. Since annotators can designate a set of proposition as support, we report two agreement measures: one for exact match and another for span overlap, in which a partial or full overlap of support span is regarded as matching. For exact match, the agreement is 57.2%, and span overlap, 75.5%. The difference is mostly due to disagreements regarding the minimal set of propositions that collectively constitute a premise. In some cases, the annotators disagree on whether a set of propositions support a given claim collectively or individually. In the former case, a single support relation is annotated with a support spanning multiple propositions, whereas in the latter case, multiple support relations are annotated, each with a single proposition as the support.

The resulting dataset consists of 731 comments, 4943 propositions, and 1282 support relations.

[7] *http://gate.ac.uk*

3.3 eRulemaking Controversy Corpus v1.0

For Step B in Figure 1.1, after loosening the requirement that support for a given claim has to appear in the same comment, we annotate support relations that exist in an entire thread of comments, where a thread is defined as a set of comments that is written in response to one another.

The corpus is comprised of user comments extracted from *Regulation-Room.org*. First, we transferred part of the annotated data from *Airline Passenger Rights* (APR) rule – a subset of a yet unpublished corpus collected at Cornell containing the relation labels of pro-arguments, or support relations.[8] The APR-Cornell dataset consists of 923 comments and 8,320 propositions (segments).

In the next step, we selected only those comments which had dialogical nature, i.e. which attracted at least one reply. This dataset was called Regulation Room Divisiveness (RRD). It consists of 209 comments, which in this case constitute turns in the dialogical exchange, and 70 maps which are graphs representing argument networks. The annotation was extended by adding more pro-arguments (using more fine-grained criteria) and con-arguments which are inherently dialogical (see Table 3.4 for the size of language resources used in this study).

Regulation Room Divisiveness Corpus

The annotation was performed using the OVA+ annotation tool [35][9] marking three types of relations between propositional contents of comments (see Table

[8] *Airline Passenger Rights* is one of the several rules comprising the corpus. See Park et al. [58] for the descriptions of the pro- relations.

[9] Available at *http://ova.arg-tech.org*

Table 3.4: Summary of the language resources: Cornell corpus for *Airline Passenger Rights* (APR) discussion; and the Regulation Room Divisiveness (RRD) corpus.

	Words	**Segments**	**Comments**	**Maps**
APR	118,789	8,320	923	-
RRD	23,682	1,657	209	70

3.5): pro-arguments (Default Inference, RA); con-arguments (Default Conflict, CA); and the relation of Rephrase (Default Rephrase, MA), which captures situations when people give the same comment, but use a different linguistic surface.

In the annotation, the argumentative function is understood as the relation between two propositions, not as the property of one span of text. In the OVA+ tool, these relations are marked as edges connecting information nodes (I-nodes) which contain propositions (see Fig. 3.1). To convert the raw text into an argument map, the analyst needs to paste the text into the left hand panel and then click on the right hand panel to create an I-node. Edges can be created by clicking the "Add edge" button and dragging the mouse between I-nodes. After the annotation, the map can be saved to the AIFdb database[10] [41] and later downloaded in various file formats (including .json and .pl). The tool is a web-based application, freely available to use for annotation of argument diagrams.

Default Inference holds between two propositions when one proposition provides a reason to accept another proposition. In other words, a supporting claim can be used to answer the question "why *p*?". In the example (1) from the map

[10] Available at *http://aifdb.org*

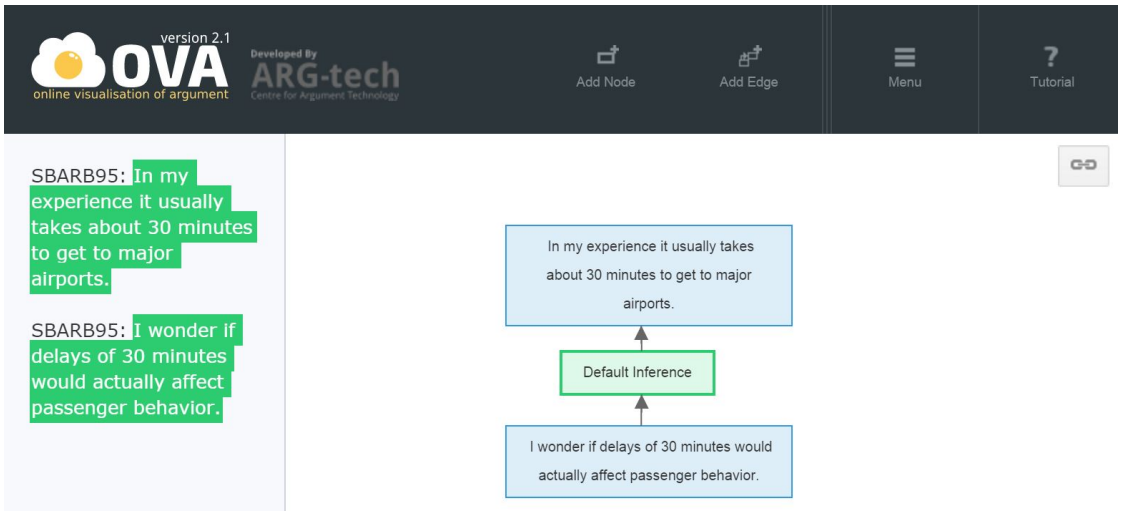

Figure 3.1: OVA+ – Online Visualisation of Arguments tool: annotation window.

RRD:#4900, (1-a) provides support for (1-b). If the propositional content of (1-a) was challenged in a dialogical situation with the question "why?", proposition (1-b) could be used as an answer to this question. In this example the user "SBARB95" is arguing that the suggested regulation (obligating airlines to inform passengers about delays longer than 30 minutes) should not be introduced. As the reason for this claim, the user "SBARB95" provides the statement that it usually takes longer than 30 minutes to travel to the airport.

(1) a. SBARB95: *In my experience it usually takes about 30 minutes to get to major airports*

b. SBARB95: *I wonder if delays of 30 minutes would actually affect passenger behavior*

Default Conflict holds between two propositions which cannot be both true at the same time. Speakers use conflicting propositions to attack another speaker's claims, by means of providing counter-claims. Example (2) from the map RRD:#4891 presents the situation in which the claim (2-a) provided by one user

is attacked with the claim (2-b) by another user. In the example, user "AK-TRAVELLER" suggests a new regulation, according to which the airlines should inform passengers in advance about possible delays or cancellations. This statement is attacked by the user "SOFIEM", who is providing a counter-claim, saying that the solution is not possible.

(2) a. AKTRAVELLER: *The airline could call in advance and give the passenger their options*
b. SOFIEM: *Unfortunately, there's no way to give advance notice*

Default Rephrase holds between two propositions with the same or similar content expressed with different linguistic surface. Our concept of Rephrase is quite broad and covers all propositions serving the same argumentative function (e.g. repeated conclusions or premises) even in cases where the meaning equivalence of the propositions is not complete. We decided to annotate the relation of Rephrase to capture the fact that rephrased content does not constitute additional support (i.e. one propositional content repeated three times should not be counted as three separate supports for a claim). In the example (3) from the map RRD:#5411) one speaker is rephrasing their own conclusion (3-a) by restating similar propositional content in (3-b). The user "DBERGER" repeats and reformulates their opinion concerning regulation on peanuts being consumed on the planes.

(3) a. DBERGER: *There must be a complete ban on tree nuts and peanuts on planes*
b. DBERGER: *Again all nuts should be banned from airplanes*

These binary annotations of relations between propositions create the

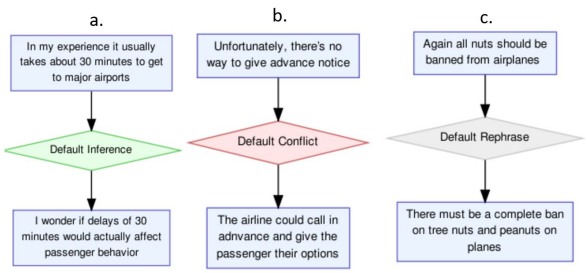

Figure 3.2: Relations of Inference, Conflict and Rephrase as building blocks of argument networks.

"building blocks" of argument networks. Results of simple annotations of examples (1), (2), (3) are presented in Fig. 3.2.

Table 3.5: Occurrences of relations between contents of comments in the annotated corpus of argument networks.

Relation type	**Number**
Inference (RA)	671
Conflict (CA)	97
Rephrase (MA)	14
Total	782

Table 3.5 presents a summary of relations of Inference, Conflict and Rephrase in the RRD corpus. The Regulation Room Divisiveness corpus is freely available at *http://arg.tech/rrd*. The corpus uses the open Argument Interchange Format (AIF) standard for argument representation [70] and constitutes a part of the AIFdb database.

In this chapter, we have presented the corpora that were built based on our theoretical model for the experiments described in the subsequent chapters.

CHAPTER 4

IDENTIFYING ELEMENTARY UNITS

In this chapter, we consider Step A in Figure 1.1. The ability to analyze the adequacy of supporting information is necessary for determining the trustworthiness of an argument.[1] This is especially the case for online user comments, which often consist of arguments lacking proper substantiation and reasoning. Thus, we develop a framework for automatically classifying each statement as UNVERIFIABLE, VERIFIABLE NONEXPERIENTIAL, or VERIFIABLE EXPERIENTIAL[2], where the appropriate type of support is *reason*, *evidence*, and *optional evidence*, respectively. Once the existing support relations among statements are identified, this classification can provide an estimate of how well the claims are supported. We use the Cornell eRulemaking - APR & MS—a dataset of 9,476 statements from 1,047 comments submitted to an eRulemaking platform (See Chapter 3—and find that Support Vector Machine (SVM) classifiers trained with n-grams and additional features capturing the verifiability and experientiality exhibit statistically significant improvement over the unigram baseline, achieving a macro-averaged F_1 score of 68.99%.

4.1 Background

Argumentation mining is a relatively new field focusing on identifying and extracting argumentative structures in documents. An *argument* is typically defined as a conclusion with supporting premises, which can be conclusions of

[1] In this work, even unsupported claims are considered part of an argument. This allows us to discuss the types of support that can further be provided to strengthen the argument, as a form of assessment.

[2] Verifiable Experiential statements are verifiable statements about personal state or experience. See Table 2.1 for examples.

other arguments themselves [78, 79, 67]. To date, much of the argumentation mining research has been conducted on domains like news articles, parliamentary records and legal documents, where the documents contain well-formed arguments, i.e. claims with supporting reasons and evidence [51, 56, 92, 25, 1].

Unlike documents written by professionals, online user comments often contain claims with inappropriate or missing justification. One way to deal with such "incomplete" arguments is to simply disregard them and focus on extracting arguments containing proper support [82, 14]. However, recognizing such statements as part of an argument,[3] and determining the appropriate types of support can be useful for assessing the adequacy of the supporting information, and in turn, the trustworthiness of the whole argument. Consider the following examples:

> *How much does a small carton of milk cost?*$_1$ *More children should drink milk*$_2$*, because children who drink milk everyday are taller than those who don't*$_3$*. Children would want to drink milk, anyway*$_4$*.*

Firstly, **Statement 1** does not need any support, nor is it part of an argument. Next, **Statement 2** is an *unverifiable* claim because it cannot be proved with objective evidence. Instead, it can be supported by a reason, as is the case in this example. If the reason, **Statement 3**, were not true, the whole argument would fall apart, giving little weight to **Statement 2**. Thus, an objective evidence supporting **Statement 3**, which is a *verifiable* statement, could be provided to strengthen the argument. Lastly, as **Statement 4** is *unverifiable*, we cannot expect an objective evidence that proves it, but a reason as its support. Note that providing

[3]Not all statements in user comments are part of an argument, e.g. questions and greetings. We address this in Section 4.4.

a reason why **Statement 3** might be true is not as effective as substantiating it with a proof, but is still better than having no support. This shows that not only the presence, but also the type of supporting information affects the strength of the argument.

Examining each statement in this way, i.e. with respect to its verifiability, provides a means to determine the desirable types of support, if any, and enables the analysis of the arguments in terms of the adequacy of their support. Thus, we propose the task of classifying each statement (the elementary unit of argumentation in this work) in an argument as UNVERIFIABLE, VERIFIABLE NONEXPERIENTIAL, or VERIFIABLE EXPERIENTIAL, where the appropriate type of support is *reason*, *evidence*, and *optional evidence*, respectively. To perform the experiments, we annotate 9,476 statements from 1,047 comments extracted from an eRulemaking platform.

4.2 Learning Algorithm

To classify each statement in an argument as VERIF$_{NON}$, VERIF$_{EXP}$, or UNVERIF, we train multiclass Support Vector Machines (SVM) as formulated by Crammer and Singer [18], and later extended by Keerthi et al.[37]. We use the LibLinear [23] implementation. We experimented with other multiclass SVM approaches such as 1-vs-all and 1-vs-1 (all-vs-all), but the differences were statistically insignificant, consistent with Hsu and Lin's [34] empirical comparison of these methods. Thus, we only report the performance of the Crammer and Singer version of Multiclass SVM.

We also formulate the classification task as a sequence labeling problem.

Each user comment consists of a sequence of propositions (in the form of sentences or clauses), and each proposition is classified based on its appropriate support type. Instead of predicting the labels individually, we jointly optimize for the sequence of labels for each comment.

Thus, we apply Conditional Random Fields (CRFs) [40] to the task. Typically, CRFs are trained in a supervised fashion. However, as labeled data is very difficult to obtain for the task of support identification, it is important to exploit distant supervision in the data to assist learning. Therefore, we investigate semi-supervised CRFs which train on both labeled and unlabeled data by using the posterior regularization (PR) framework [28]. PR has been successfully applied to many structured NLP tasks such as dependency parsing, information extraction and sentiment analysis tasks [27, 2, 93].

The training objective for semi-supervised CRFs augments the standard CRF objective with a posterior regularizer:

$$\begin{aligned} \max_{\theta} \mathcal{L}(\theta) - \min_{q \in Q} \{ & KL(q(\mathbf{Y}) \| p_\theta(\mathbf{Y}|\mathbf{X})) \\ & + \beta \| E_q[\phi(\mathbf{X}, \mathbf{Y})] - \mathbf{b} \|_2^2 \} \end{aligned} \tag{4.1}$$

The idea is to find an optimal auxiliary distribution q that is close to the model distribution $p_\theta(\mathbf{Y}|\mathbf{X})$ (measured by KL divergence) which satisfies a set of posterior constraints. We consider equality constraints which are in the form of $E_q[\phi(\mathbf{X}, \mathbf{Y})] = \mathbf{b}$, where $\mathbf{b}$ is set based on domain knowledge. We can also consider these constraints as features, which encode indicative patterns for a given support type label and prior beliefs on the correlations between the patterns and the true labels.

In this work, we consider two ways of generating constraints. One approach is to manually define constraints, leveraging on our domain knowledge. For

instance, the unigram "should" is usually used as part of imperative, meaning it is tightly associated with the UNVERIF class. Similarly, having 2 or more occurrences of a strong subjective token is also a distinguishing feature for UNVERIF. We manually define 10 constraints in this way. The other approach is to automatically extract constraints from the given labeled training data using information gain with respect to the classes as a guide.

4.3 Features

The features are binary-valued, and the feature vector for each data point is normalized to have the unit length: "Presence" features are binary features indicating whether the given feature is present in the statement or not; "Count" features are numeric counts of the occurrence of each feature is converted to a set of three binary features each denoting 0, 1 and 2 or more occurrences. We first tried a *binning* method with each digit as its own interval, resulting in binary features of the form $featCnt_n$, but the three-interval approach proved to be better empirically, and is consistent with the approach by Riloff and Shoen [72].

The features can be grouped into three categories by purpose: Verifiability-specific (VER), Experientiality-specific (EXP) and Basic Features, each designed to capture the given statement's verifiability, experientiality, and both, respectively. Now we discuss the features in more detail.

Basic Features

N-gram Presence A set of binary features denote whether a given unigram or bigram occurs in the statement. The intuition is that by examining the occurrence of words or phrases in VERIF$_{NON}$, VERIF$_{EXP}$, and UNVERIF statements, the classes that have close ties to certain words and phrases can be identified. For instance, when a statement contains the word *happy*, the statement tends to be UNVERIF. From this observation, we can speculate that *happy* is highly associated with UNVERIF, and *went*, VERIF$_{EXP}$. n-gram presence, rather than the raw or normalized frequency is chosen for its superior performance [55].

Core Clause Tag (CCT) To correctly classify statements with main or subordinate clauses that do not affect the verifiability of the statement (e.g. statements 8 through 10 in Table 2.1, respectively), it is necessary to distinguish features that appear in the main clause from those that appear in the subordinate clause. Thus, we employ an auxiliary feature that adds clausal information to other features by tagging them as either *core* or *accessory* clause.

Let's consider statements 7, 9 and 10 in Table 2.1:

To realize this intuition, we use syntactic parse trees generated by the Stanford Parser [20]. In particular, Penn Treebank 2 Tags contain a clause-level tag *SBAR* denoting a "clause introduced by a subordinating conjunction" [49]. The "that" clause in statement 10 spans a subtree rooted by *SBAR*, whose left-most child has a lexical value "that." Similarly, the subordinate (non-italicized) clause in statement 9 falls in a subtree rooted by *SBAR*, whose only child is *S*. Once the main clause of a given statement is identified, all features set off by the clause are tagged as "core" and the rest are tagged as "accessory." If a speech event is

present, the tags are flipped.

Verifiability-specific Features (*VER*)

Parts-of-Speech (POS) Count Rayson et al. [71] have shown that the POS distribution is distinct in imaginative vs. informative writing. We expect this feature to distinguish UNVERIF statements from the rest.

Sentiment Clue Count Wilson et al. [88] provides a subjectivity clue lexicon, which is a list of words with sentiment strength tags, either strong or weak, along with additional information, such as the sentiment polarity, *Part-of-Speech Count* (POS), etc. We suspect that statements containing more sentiment words is more likely to be UNVERIF.

Speech Event Count We use the 50 most frequent *Objective-speech-event* text anchors crawled from the *MPQA 2.0* corpus [89] as a speech event lexicon. The speech event text anchors refer to words like "stated" and "wrote" that introduce written or spoken statements attributed to a source. Statements containing speech events such as statement 10 in Table 2.1 are generally VERIF_{NON} or VERIF_{EXP}, since whether the attributed source has indeed made the statement he allegedly made is objectively verifiable regardless of the subjectivity of the statement itself.

Imperative Expression Count Imperatives, i.e. commands, are generally UNVERIF (e.g. "Do the homework now!" that is, we expect there to be no objective evidence proving that the homework should be done right away.), unless the sentence is a law or general procedure (e.g. "The library should allow you to check out books." where the context makes it clear that the writer is claiming

that the libary lends out books.) This feature denotes whether the statement begins with a verb or contains the following: *must, should, need to, have to, ought to*.

Emotion Expression Count These features target specific tokens "!", and "..." as well as fully capitalized word tokens to capture the emotion in text. The rationale is that expression of emotion is likely to be more prevalent in UNVERIF statements.

Experientiality-specific Features (*EXP*)

Tense Count Statements written in past tense are rarely VERIF_{NON}, because even in the case that the statment is verifiable, they are likely to be the commenter's past experience, i.e. VERIF_{EXP}. Future tense are typically UNVERIF because claims about what will happen in the future are often unverifiable with objective evidence, with exception being statements like predictions based on policy of organizations, i.e. "Fedex will deliver on Sunday." To take advantage of these observations, three binary features capture each of three tenses: *past*, *present*, and *future*.

Person Count First person narratives can suggest that the statement is UNVERIF or VERIF_{EXP}, except for rare cases like "We, the passengers,..." in which the first person pronoun refers to a large body of individuals. This intuition is captured by having binary features for: *1st, 2nd* and *3rd person*.

Regulation	VERIF$_{NON}$	VERIF$_{EXP}$	UNVERIF	Subtotal	OTHER	Total	# of Comments
APR	1106	851	4413	6370	522	6892	820
HMCP	251	416	1733	2400	186	2586	227
Total	1357	1267	6146	8770	708	9476	1047

Table 4.1: Class Distribution Over Statements

4.4 Experimental Setup

A Note on Argument Detection A natural first step in argumentation mining is to determine which portions of the given document comprise an argument. It can also be framed as a binary classification task in which each statement in the document needs to be classified as either argumentative or not. Some authors choose to skip this step [25], while others make use of various classifiers to achieve high level of accuracy, as Palau and Moens achieved over 70% accuracy on Araucaria and ECHR corpus [16, 56].

Our setup is a bit unique in that we have a notion of an "incomplete" argument, where a claim or opinion that is or could be supported with evidence or reason are considered argumentative. As a result, only about 7% ($\frac{\text{OTHER}}{\text{TOTAL}}$ in Table 4.1) of our entire dataset is marked as non-argumentative, most of which consists of questions and greetings. By simply searching for specific unigrams, such as "?" and "thank", we achieve over 99% F_1 score in determining which statements are part of an argument.

The remaining experiments were done without non-argumentative statements, i.e. OTHER in Table 4.1.

Setup We first randomly selected 292 comments as held-out test set, resulting in the distribution shown in Table 4.2. Then, VERIF$_{NON}$ and VERIF$_{EXP}$ in the

training set were oversampled so that the classes are equally distributed. During training, five fold cross-validation was done on the training set to tune the *C* parameter to *32*. Because the micro-averaged F_1 score can be easily boosted on datasets with highly skewed class distribution, we optimize for the macro-averaged F_1 score.

Preprocessing was kept at a minimal level: capital letters were lowercased after counting fully capitalized words, and numbers were converted to a *NUM* token.

4.5 Results & Analysis

4.5.1 Multiclass-SVM

Table 4.3 shows a summary of the classification results. The best overall performance is achieved by combining all features (*UNI+BI+VER+EXP*), yielding 68.99% macro-averaged F_1, where the gain over the baseline is statistically significant according to the bootstrap method with 10,000 samples [22, 5].

Core Clause Tag (CCT) We do not report the performance of employing feature sets with *Core Clause Tag (CCT)* in Table 4.3, because the effect of *CCT* on

	VERIF$_{NON}$	VERIF$_{EXP}$	UNVERIF	Total
Train	987	900	4459	6346
Test	370	367	1687	2424
Total	1357	1267	6146	8770

Table 4.2: # of Statements in Train and Test Set

Features	UNVERIF vs All			VERIF$_{NON}$ vs All			VERIF$_{EXP}$ vs All			Average F_1	
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Macro	Micro
UNI(base)	85.24	79.43	82.23	42.57	51.89	46.77	61.10	66.76	63.80	64.27	73.31
UNI+BI	82.14	89.69*	85.75*	51.67*	37.57	43.51	73.48*	62.67	67.65*	65.63	77.64*
VER	88.52*	52.10	65.60	28.41	61.35*	38.84	42.41	73.02*	53.65	52.70	56.68
EXP	82.42	4.45	8.44	20.92	76.49*	32.85	31.02	82.83*	45.14	28.81	27.31
VER+EXP	89.40*	49.50	63.72	29.25	71.62*	41.54	50.00	79.56*	61.41	55.55	57.43
UNI+BI+ VER+EXP	86.86*	83.05*	84.91*	49.88*	55.14	52.37*	66.67*	73.02*	69.70*	**68.99***	77.27*

Table 4.3: Three class classification results in % (Crammer & Singer's Multiclass SVMs)

Precision, recall, and F_1 scores are computed with respect to each one-vs-all classification problem for evaluation purposes, though a single machine is built for the multiclass classification problem, instead of 3 one-vs-all classifiers. The star (*) indicates that the given result is statistically significantly better than the unigram baseline.

each of the six sets of features is statistically insignificant. This is surprising at first, given the strong motivation for distinguishing the core clause from auxiliary clause, as addressed in the previous section: Subordinate clauses like "I believe" should not cause the entire statement to be classified as UNVERIF, and clauses like "He said" should serve as a queue for VERIF$_{NON}$ or VERIF$_{EXP}$, even if an unverifiable clause follows it. Our conjecture turned out to be wrong, mainly because such distinction can be made for only a small subset of the data: For instance, over 83% of the unigrams are tagged as *core* in the *UNI* feature set. Thus, most of the important features for feature sets with *CCT* end up being features with *core* tag, and the important features for feature sets with and without *CCT* are practically the same, as shown in Table 4.4, resulting in statistically insignificant performance differences.

Informative Features The most informative features reported in Table 4.5 exhibit interesting differences among the three classes: Sentiment bearing words,

Feats		*UNI*	UNI_{CCT}
UNVERIF	+	should, whatever, responsibility	$should_C$, $should_A$, $understand_C$
	-	previous, solve, florida, exposed, reacted, reply, kinds	$exposed_C$, $solve_C$, NUM_C, $florida_C$, $reacted_C$, $pool_C$, $owed_C$
$VERIF_{NON}$	+	impacted, NUM, solve, cars, pull, kinds, congress	$impacted_C$, $solve_C$, $cars_C$, NUM_C, $pool_C$, $writing_C$, $death_C$, $link_C$
	-	should, seems, comments	$should_C$, $comments_C$
$VERIF_{EXP}$	+	owed, consumed, saw, expert, interesting, him, reacted, refinance	$owed_C$, $consumed_C$, $expert_C$, $reacted_C$, $happened_C$, $interesting_C$
	-	impacted, wo	$impacted_C$, wo_C, $concern_C$, $died_C$

Table 4.4: Most Informative Features for *UNI* and UNI_{CCT}

10 Unigrams with the largest weight (magnitude) with respect to each class (+ : positive weight / - : negative weight).

Feature Set		*UNI+BI+VER+EXP*
UNVERIF	+	should,$StrSentClue_{>2}$, $VB_{>2}$
	-	$StrSentClue_0$, $VBD_{>2}$, air, since, no_one, allergic, not_an
$VERIF_{NON}$	+	die, death, reaction, person, allergen, airborne, no_one, allergies
	-	PER_{1st}, should
$VERIF_{EXP}$	+	$VBD_{>2}$, PER_{1st}, i_have, his, he, him, time_!
	-	$VBZ_{>2}$, PER_{2nd}

Table 4.5: Most Informative Features for *UNI+BI+VER+EXP*

10 Features with the largest weight (magnitude) with respect to each class (+ : positive weight / - : negative weight).

i.e. "should" and strong sentiment clues, are good indicators of UNVERIF, whereas person and tense information is crucial for $VERIF_{EXP}$. As expected, the strong indicators of UNVERIF and $VERIF_{EXP}$, namely "should" and PER_{1st} are negatively associated with $VERIF_{NON}$. It is intriguing to see that the heavily weighted features of $VERIF_{NON}$ are non-verb content words, unlike those of the other classes. One explanation for this is that $VERIF_{NON}$ are rarely indicated by specific cues; instead, a good sign of $VERIF_{NON}$ is the absences of cues for the other classes, which are often function words and verbs. What is remaining, then, are non-verb content words. Also, certain content words are more likely to bring about factual discussions. For instance, technical terms like"allergen"

and "airborne," appear in verifiable non-experiential statements as "The FDA requires labeling for the following 8 allergens."

Non-n-gram Features Table 4.3 clearly shows that the three non-n-gram features, *VER*, *EXP*, and *VER+EXP*, do not perform as well as the n-gram features. But still, the performance is impressive, given the drastic difference in the dimensionality of the features: Even the combined feature set, *VER+EXP*, consists of only about 100 features, when there are over 8,000 unigrams and close to 70,000 bigrams. In other words, the non-n-gram features are effectively capturing characteristics of each class. This is very promising, since this shows that a better understanding of the types of statement can potentially lead to a more concise set of features with equal, or even better, performance.

Also notice that *VER* outperforms *EXP* for the most part, even with respect to VERIF_{NON} *vs All* and VERIF_{EXP} *vs All*, except for recall. This is intriguing, because *VER* are mostly from subjectivity detection domain, intended to capture the subjectivity of words in the statements leveraging on pre-built lexia. Simply considering subjectivity of words should provide no means of distinguishing VERIF_{NON} from VERIF_{EXP}. One of the reasons for *VER*'s superior performance over *EXP* is that *EXP* by itself is inadequate for the classification task: *EXP* consists of only 6 (or 12 with CCT) features denoting the person and tense information. Another reason is that *VER*, in a limited fashion, does encode experientiality: For instance, past tense statements can be identified with the existence of *VBD*(verb, past tense) and *VBN*(verb, past participle).

Word pairs as features. Starting with earlier works that proposed them as features [48], some form of *word pairs* has generally been part of feature sets for implicit discourse relation recognition. According to our research, however, these

features provide little or no additional gain, once other features are employed. This seems sensible, since we now have a clearer idea of the types of information important for the task and have developed a variety of feature types, each of which aims to represent these specific aspects of the discourse relation arguments. Thus, general features like *word pairs* may no longer have a role to play for implicit discourse relation identification.

Preprocessing. Preprocessing turned out to impact the classifier performance immensely, especially for features like *polarity* and *inquirer tags* that rely on information retrieved from a lexicon. For these features, if a match for a given word is not found in the lexicon, no information is passed on to the classifier.

As an example, consider the General Inquirer lexicon. Most of its verb entries are present tense singular in form; thus, without stemming, dictionary look up fails for a large portion of the verbs. In our case, the F_1-score increases by roughly 10% after stemming.

Further tuning is possible by a few hand-written rules to guide lexicon lookup. The word *supplied*, for instance, becomes *suppli* after stemming, which still fails to match the lexicon entry *supply*, unless adjusted accordingly.

Binning. An additional finding regards features that capture numeric, rather than binary, information, such as *polarity*. Since this feature encodes the counts of each type of sentiment words (with respect to each argument and their cross product), and Naive Bayes can only interpret binary features, we first employed binning mechanism with each bin covering a single value. For instance, if arg1 consists of three positive words, we included *arg1pos1*, *arg1pos2* and *arg1pos3* as features instead of just *arg1pos3*.

The rationale behind binning is that it captures the proximity of related instances. Imagine having three instances each with one, two, and three positive words in arg1, respectively. Without binning, the features added are simply *arg1pos1*, *arg1pos2*, *arg1pos3*, respectively. From the perspective of the classifier, the third instance is no more similar to the second instance than it is to the first instance, even though having three positive words is clearly closer to having two positive words than having one positive word. With binning, this proximity is captured by the fact that the first instance has just one feature in common with the third instance, whereas the second instance has two.

Binning, however, results in single digit F_1-scores on most of the classification tasks. Without binning, the performance increases significantly. One possible explanation is that these features function as an abstraction of certain lexical patterns, rather than directly capturing similarities among the instances of the same class.

4.5.2 CRF

Method	UNVERIF vs All			VERIF$_{NON}$ vs All			VERIF$_{EXP}$ vs All			F_1
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	(Macro-Ave.)
Multi-SVM (P&C)	**86.86**	83.05	84.91	49.88	**55.14**	**52.37**	66.67	**73.02**	**69.70**	**68.99**
Super-CRF 100%	80.35	**93.30**	**86.34**	**60.34**	28.38	38.60	**74.57**	59.13	65.96	63.63

Table 4.6: Multi-SVM vs Supervised CRF Classification Results

CRF vs Multiclass SVM As shown in Table 4.6, the multiclass SVM classifier performs better overall. But at the same time, a clear trend can be observed: With CRF, the precision makes a significant gain at the cost of the recall for both VERIF$_{NON}$ and VERIF$_{EXP}$. And the opposite is the case for VERIF.

Method	UNVERIF vs All			VERIF$_{NON}$ vs All			VERIF$_{EXP}$ vs All			F_1
	Pre.	Rec.	F_1	Pre.	Rec.	F_1	Pre.	Rec.	F_1	(Macro-Ave.)
Super-CRF 100%	80.35	93.30	86.34	60.34	28.38	38.60	74.57	59.13	65.96	63.63
Super-CRF 75%	79.57	92.59	85.59	54.33	30.54	39.10	77.08	53.13	62.90	62.53
CRF-PR$_H$ 75%	79.42	93.12	85.73	57.14	31.35	40.49	79.01	52.32	62.95	63.06
CRF-PR$_{H+IG}$ 75%	79.72	94.37	86.43	63.58	27.84	38.72	76.6	55.31	64.24	63.13
Super-CRF 50%	79.16	93.01	85.53	51.92	21.89	30.82	71.68	55.86	62.79	59.71
CRF-PR$_H$ 50%	79.28	92.12	85.17	55.68	26.49	35.92	69.23	53.95	60.64	60.57
CRF-PR$_{H+IG}$ 50%	79.23	92.23	85.24	55.37	26.49	35.83	70.32	54.22	61.23	60.77
Super-CRF 25%	75.93	96.86	85.13	57.89	5.95	10.78	79.06	50.41	61.56	52.49
CRF-PR$_H$ 25%	76.27	96.03	85.02	41.54	7.30	12.41	79.15	50.68	61.79	53.07
CRF-PR$_{H+IG}$ 25%	75.83	96.32	84.86	38.78	5.14	9.07	79.31	50.14	61.44	51.79

Table 4.7: Supervised vs Semi-Supervised CRF Classification Results

*The percentages refer to the percentages of the labeled data in the training set.

*The methods are as follows: Super-CRF = supervised approach only using the labeled data, CRF-PR$_H$ = CRF with posterior regularization using constraints that are manually selected, CRF-PR$_{H+IG}$ = CRF with posterior regularization using constraints that are manually written and automatically generated using information gain.

*Precision, recall, and F_1 scores are computed with respect to each one-vs-all classification problem for evaluation purposes, though a single model is built for the multi-class classification problem.

One cause for this is the heavy skew in the dataset that can be better handled in SVMs; As mentioned before, the majority class (UNVERIF) comprises about 70% of the dataset. When training the multiclass SVM, it is relatively straight forward to balance the class distribution in the training set, as each proposition is assumed to be independent of others. Thus, we randomly oversample the instances of non-majority classes to construct a balanced trained set. The situation is different for CRF, since the entire sequence of propositions comprising a comment is classified together. Further investigation in resolving this issue is desirable.

Semi-supervised CRF Table 4.7 reports the average performance of CRFs trained on 25%, 50%, 75% and 100% labeled training data (the same dataset), using various supervised and semi-supervised approaches over 5 rounds. Though, the amount is small, incorporating semi-supervised approaches consistently boosts the performance for the most part. The limited gain in performance is due to the small set of accurate constraints.

One crucial component of training CRFs with Posterior Regularization is designing constraints on features. For a given feature, a respective constraint defines a probability distribution over the possible classes. For the best performance, the distribution needs to be accurate, and the constrained features occur in the unlabeled training set frequently.

Our manual approach resulted in a small set of about 10 constraints on features that are tightly coupled with a class. Examples include the word "should", large number of strong subjective expressions, and imperatives, which are all highly correlated with the UNVERIF. While the constraints are accurate, the coverage is too small to boost the performance. However, it is quite difficult to generate a large set of constraints, because there are not that many features that are indicative of a single class. Also, given that UNVERIF comprises a large percentage of the dataset, and the nature of verifiability[4], it is even more difficult to identify features tightly coupled with VERIF_{NON} and VERIF_{EXP} class. One issue with automatically generated constraints, based on information gain, is that they tend to be inaccurate.

[4]Verifiability does not have many characterizing features, but the lack of any of the characteristics of unverifiability, such as sentiment bearing words, is indicative of verifiability.

4.6 Conclusions

We have proposed a novel task of automatically classifying each statement as UNVERIFIABLE, VERIFIABLE NONEXPERIENTIAL, or VERIFIABLE EXPERIENTIAL, where the appropriate type of support is *reason*, *evidence*, and *optional evidence*, respectively. This classification, once the existing support relations among statements are identified, can provide an estimate of how well the claims are supported. We find that SVMs trained with n-grams and other features to capture the verifiability and experientiality exhibit statistically significant improvement over the unigram baseline, achieving a macro-averaged F_1 score of 68.99%. While the overall performance is reduced, we find that CRFs improves the F_1 score with respect to the UNVERIF class. Also, semi-supervised CRFs with posterior regularization trained on 75% labeled training data can closely match the performance of a supervised CRF trained on the same training data with the remaining 25% labeled as well.

One avenue for future work is to incorporate the identification of relations among the statements in an argument to the system to analyze the adequacy of the supporting information in the argument. This can be used to recommend comments to readers and provide feedback to writers so that they can construct better arguments. An efficient way to handle the skewed distribution of classes in the training set is needed to boost the performance of CRFs. And a set of efficient constraints is necessary for better performing semi-supervised CRFs with posterior regularization.

In the next chapter, we present how support relations among the elementary units can be automatically identified.

CHAPTER 5

IDENTIFYING SUPPORT RELATIONS

In this chapter, we consider Step B in Figure 1.1, after loosening the requirement that support for a given claim needs to be found within the comment. More specifically, we consider support relations that exist in an entire thread of comments, where a thread is defined as a set of comments that are written in response to one another.

5.1 Background

In the age of vast amounts of information being communicated through the Internet, it is not surprising that political dialogue is increasingly taking place online too. Governments around the world are increasingly utilising online platforms and social media in order to engage with, and ascertain the opinions of, their citizens [33, 53]. Whilst policy makers could potentially benefit from such feedback from society, they first face the challenge of making sense out of the large volumes of data produced. Identifying those issues which are key to the debate, those which are the most controversial, those which were successfully resolved, and those which should be further handled to achieve consensus and mutual understanding, is a skilled and time-consuming task in real-life discussions.

Our goal here is to combine a variety of techniques, some based on general linguistic features, others on features that are specific to argumentation, in order to automate the task of identifying the structure of the arguments and how they interconnect in a broader discussion. Though this task is extremely demand-

ing for current text mining and computational linguistics techniques, our final target is not the argumentative structures themselves, but rather argumentative structures which are sufficiently accurate to develop an *interpretative* step that gives decision-makers some insight into the discussion. Here we use the simple metric of centrality, and show that even with modest performance on the task of extracting the argument network, it is possible to generate rather high reliability in identifying central issues to the discussion.

In the remainder of the chapter, we look at automating the argument analysis task. More specifically, we build classifiers to distinguish propositions in support relation from those that are not. The task is formulated as identifying proposition pairs (ordered) in support relation from all possible ordered pairs of propositions in a given thread. By using argument mining techniques to produce the kind of argumentative structures which we are able to obtain manually, it would be possible to give a real time overview of the state of a particular debate, and thus allowing for interactions with the debate in order to resolve controversial issues, or pursue topics that are central, as they arise. Starting with manually segmented text, we then consider three techniques: firstly, we use topical similarity in order to reduce the possible search space of connected propositions, we then look at identifying discourse indicators, strong lexical cues indicating the role of a proposition in the dialogue, and finally, we apply computational discourse analysis techniques to identify the connections between propositions.

5.2 Reducing the Search Space

Our corpus (See Chapter 3 contains over 1,500 segments across 70 nodesets, corresponding to individual threads in the dialogue, resulting in over 20,000 potential connections between segments in the same nodeset. Our first step is to reduce the size of the search space. We do this using semantic similarity to determine those propositions which are discussing similar topics. This method is similar to that presented in [43], where it is assumed firstly that the argument structure to be determined can be represented as a tree, and secondly, that this tree is presented depth first. That is, the conclusion is given first and then a line of reasoning is followed supporting this conclusion. Once that line of reasoning is exhausted, the argument moves back up the tree to one of the previously made points.

Based on these assumptions it is possible to determine connections by looking at how semantically similar each proposition is to its predecessor. If they are similar, then we assume that they are connected and the current line of reasoning is being followed. If they are not sufficiently similar, then we first consider whether we are moving back up the tree, and compare the current proposition to all of those made previously and, if the most similar previous point is above a set threshold, we connect them. Finally, if the current point is not related to any of those made previously, then it is assumed that a new topic is being discussed, and the proposition is left, unconnected, as the root of this new argument.

We exploit the dialogical structure of our data by discounting any possible connections between propositions which are not in the same thread of the dialogue. The way that a connection is determined also gives precedence to

connections between adjacent propositions in the same comment. Two different thresholds are used, a lower threshold for sequential propositions which are more likely to be connected, and a higher threshold for non-sequential propositions. In all cases, the threshold values were selected to maximise recall, whilst keeping precision at a reasonable level. This trade-off was made as our goal is to narrow the search space, reducing the number of possible pairs as much as possible whilst losing a minimum number of connected pairs.

The first approach which we consider uses WordNet[1] to determine the similarity between the synsets of each word in the first proposition and each word in the second. This relatedness score is inversely proportional to the number of nodes along the shortest path between the synsets. The shortest possible path occurs when the two synsets are the same, in which case the length is 1, and thus, the maximum relatedness value is 1. We then look at the maximum of these values in order to pair a word in the first proposition to one in the second. From here we then considered a range of different methods to determine whether the two propositions are connected:

1. **Average score:** takes the sum of the scores for each pairing and divides by the total number of paired words.
2. **Maximum score:** looks only at the pairing with the greatest score.
3. **Average of top two scores:** takes the average of the scores for the two most similar words.
4. **Average of top three scores:** takes the average of the scores for the three most similar words.

[1] *http://wordnet.princeton.edu/*

5. **Weighted average score:** takes the average score for each pairing, giving a higher weight to the most similar, and then reducing this weighting as the similarity decreases.

The average precision, recall and F-score obtained using each of these possible methods is shown in Table 5.1.

We also implemented two further methods of determining connectedness using semantic similarity. The first of these approaches uses word2vec [50] to train each word as a vector with 200 dimensions. Each proposition can then be represented as a set of vectors corresponding to the words in the proposition:

$$W_i = (w_t, w_{t+1}, ...w_{t+k})$$

where w_t is a word vector, and W_i is the vector set representing a proposition. The vector sets of two propositions, W_i and W_j, can then be used form a matrix $M_{i,j}$:

$$M_{i,j} = W_i W_j^T = \begin{bmatrix} w_t w_v & \cdots & w_t w_{v+l} \\ \vdots & \ddots & \vdots \\ w_{t+k} w_v & \cdots & w_{t+k} w_{v+l} \end{bmatrix}$$

where $(w_t w_v)$ is the cosine similarity of w_t and w_v. We are then able to calculate the similarity of $proposition_i$ and $proposition_j$ as follows:

$$Sim_{i,j} = \frac{\sum_{m=i,n=j} max(M_{m,n})}{\sqrt{length_i length_j}}$$

Again we used a threshold of similarity values chosen to maximise recall, and the results obtained can be seen in Table 5.1.

The final approach which we implemented used doc2vec [44] to represent every proposition as a vector with 200 dimensions, and then calculated the cosine similarity between vectors to represent the proposition similarity.

In each case, as our aim here is simply to reduce the search space, the threshold values were lowered to maximise recall and so reduce the number of possible connections whilst retaining the greatest number of those propositions which had been identified as connected in the manual analysis. These results can be seen in Table 5.1, compared to a baseline obtained by assuming that each sequential proposition is connected. Despite the approaches tested giving overall lower accuracy than the baseline, we were able, in each case, to obtain a higher value for the recall. Although each method resulted in a similar level of accuracy, the average score performed best, as such we used the pairs of connected propositions obtained by this method as input to the classifiers described in Section 5.4, reducing the number of possible connections by 17.5%.

5.3 Determining Discourse Indicators

Discourse indicators are words which serve as a clue for the argumentative function of the proposition. They can either connect two propositions (inter-proposition indicators) or constitute part of the proposition (intra-proposition indicators). Certain indicators have been listed in the literature (see Table 5.6 for an aggregate list). In order to further broaden this list, we used keyword method in certain sub-sets of eRCC corpus. The indicators discovered in this

Table 5.1: Comparison of different methods for reducing the search space by determining connectedness using semantic similarity, optimised for maximum recall

Method	**Precision**	**Recall**	**F1**
Average score	0.17	**0.92**	0.29
Maximum score	0.17	0.90	0.29
Average of top two scores	0.17	0.90	0.29
Average of top three scores	0.17	0.89	0.28
Weighted average score	0.18	0.88	0.30
word2vec	0.15	0.83	0.25
doc2vec	0.12	0.85	0.21
Sequential Baseline	*0.38*	*0.65*	*0.48*

step were then used as features for the classifier discussed in Section 5.4.

A keyword is a word which has much higher frequency in one corpus than in other and the keyness of a given word indicates its overuse in one corpus as compared to other corpus [74]. The corpus for which the overuse is determined (source corpus) is compared with the reference corpus. We created 12 sub-corpora of propositions holding certain argumentative function[2]. By looking at these subcorpora separately, we are able to determine those words which, for example, are more commonly found in an Attacking proposition than in a proposition which does not attack any of the others.

For each of the sub-corpora, keywords were extracted using Log Likelihood method (threshold of critical value = 3.84, $p < 0.05$). This allowed for the determination of the list of words overused in propositions holding certain ar-

[2]Supporting, Supported, Attacking, Attacked, ERExample Prem, ERExample Concl, ERExpertOpinion Prem, ERExpertOpinion Concl, ERPracticalReason Prem, ERPracticalReason Concl, ERAdHominem Attacking, ERAdHominem Attacked

gumentative function. From the list of obtained keywords, words which were topic-specific (such as "allergy", "children", "airplane") were removed. From the total of 12 corpora comparison, only 6 brought relevant results (i.e. results both statistically significant and topic-independent). The resulting list of keywords (presented in Table 5.6) indicates words specific for this type of discourse (online comments on legal regulations) which indicate propositions with certain argumentative functions. The rationale for the choice of these words is as follows:

1. Indicating Attacking:

    - *argument*: the users of the forum use the word "argument" in attacking, rather than in any other argumentative move, as a metadiscourse marker, in some ways announcing that they are about to attack someone's argument, as in examples:"This slippery slope argument is a false one", "Your argument is a strawman".
    - *you, your*: are specific for attacking moves, due to the personal engagement and AdHominem nature of many attacks, as in example: "If you have a problem, it is up to you to have the solution"
    - *negative words (funeral, death)*: due to the emotional nature of the forum discussion, users refer to negative consequences and use hyperbole to make their attack look stronger: "But some of the people on this board calling for funerals before advancing the discussion give new meaning to the Founders' fears of the tyranny of the majority"

2. Indicating Supported

    - *should*: due to the nature of the discussion (proposition of new legal regulations), propositions expressed in deontic modality were ex-

pressed by users, and were more often used as premises (in our annotation: supported) than conclusions: "A similar problem, that should also be addressed, along with the peanut allergy problem, is the case of allowing small domestic pets in the cabin of a aircraft."

- *I think*: this bigram is used as a hedge, lowering the level of confidence the speaker ascribes to the truth of the proposition; taken into account that in argument, asserting the truth of the conclusion cannot be stronger than asserting the truth of its weakest premises, it is not surprising that users of the forum were hedging conclusions but not premises: "I think a ban of all peanuts and nuts (or at least peanuts) would be the safest route for those with peanut allergies."

In our new approach to the indicators, we broaden the concept of lexical indicators. We assume not only connectives between propositions but also specific lexical items (unigrams, bigrams) which appear inside the phrase. It could be hard to indicate certain and not topic-specific lexical indicators or constructions for argument structure in general, but it is possible to show specific lexical features of certain argumentative schemes. For example, in ERExample speakers use *I/me* and action verbs and in ERExpertOpinion we can expect a Named Entity to be present. A full list of intra-proposition discourse indicators can be seen in Table 5.6. Some of those identified are probably genre-specific (and specific for American English), but, we expect not topic-specific.

Intra-proposition discourse markers work not only for consecutive propositions but also for any propositions which are topically related (e.g. Ad Hominem attack may refer to the proposition of a person speaking many turns before).

In order to determine the validity of the identified indicators, we performed classification of propositions based on their presence, obtaining a precision of 0.82, recall of 0.19 for support relations and precision of 0.73, recall of 0.14 for attack relations. Although in both cases, the precision is high, the fact that these types of indicators are often omitted means that they do not give a good indication of the argumentative structure on their own. However, when they do occur, they give a very strong indication of the role that a proposition is playing in the dialogue, and, as such, provide a useful feature for the machine learning technique discussed in the next section.

5.4 Classifying Relations Between Propositions

This component is the final step of the automation process in which propositions in support relations are identified. As previously mentioned, the task is formulated as identifying ordered propositions pairs in support relation, i.e. the first proposition in the pair supports the second. The number of all possible ordered pairs of propositions is quadratic to the number of propositions in a given thread. Since the vast majority of them are not in support relation there is a significant imbalance in the class distribution. Thus, we only consider the proposition pairs that are classified as topically similar during search space reduction described in section 5.2. This precisely why the search space reduction was optimised for recall.

Setup. We adopt a general approach in computational discourse analysis where classification algorithms, such as Support Vector Machines (SVM) and Naive

Bayes, are used with various lexical and syntactic features [59].[3] The main difference is that traditional discourse analysis in NLP focuses on a broader set of relations, such as contingency, comparison, expansion, and temporal, whereas only support relation is targeted in this work. Also, in previous work using Penn Discourse Treebank [69], only adjacent text spans are considered, while we aim to deal with relations between propositions that may not be adjacent to each other. Because of this difference, the most informative features for this task are dissimilar to those for discourse analysis, though all the features have previously been employed in discourse analysis. In addition to the machine learning approach, we also report results using hand-coded rule-based classifier that returns true if the given pair of propositions are adjacent and contain at least 1 discourse marker, and returns false, otherwise.

Below are brief descriptions of features whose efficacy have been empirically determined in prior work[4], along with the rationale behind them:

- **Word Pairs** is the Cartesian product of the unigrams from proposition 1 with those from proposition 2. Word pairs can potentially capture semantic support relations, e.g. between "rain" and "wet." To elaborate, with enough occurrences of proposition pairs annotated as support where "rain" appears in the first and "wet" appears in the second, the model will learn that there is a support relation between "rain" and "wet." More generally, the intuition is that indicators of support relation should exist in both propositions under consideration, since we also consider propositions that are not adjacent to each other. And word pairs is an extension

[3] Laplacian Smoothing was used for Naive Bayes, and SVM was training with linear kernel where the hyper-parameters were tuned through cross-validation.

[4] Word Pairs [48], First-Last-First3 [87], Verbs [65], and Production Rules [45].

of unigrams to tasks involving pairs of propositions. Note that, while discourse connectives, such as "because," are strong indicators of support relations, they are only applicable to proposition pairs that are adjacent.

- **First-Last-First3** is the first, last, and first three words of proposition 1 and those of 2. The goal is to capture discourse indicators, or expressions that function as discourse indicators. Even when a known list of discourse indicators, such as *because, since*, and *therefore* is used as a feature, First-Last-First3 can be useful, as it also captures multiword expressions such as "as a result."

- **Verbs** is the count of pairs of verbs from proposition 1 and proposition 2 belonging to the same Levin English Verb Class [54]; the average lengths of verb phrases as well as their Cartesian product; and lastly, the part of speech of the main verb from each argument. Levin Verb classes provide a means of clustering verbs according to their meanings and behaviors. Also, longer verb phrases may indicate support in the form of justification.

- **Production Rules** refers to three features denoting the use of syntactic production rules in proposition 1, proposition 2 or both. The syntactic structure of an argument can influence that of the other argument as well as its relation type. We take the smallest units of the syntactic parse trees, i.e. production rules, as features to minimise the sparsity problem. A parse tree consists of applications of production rules, such as "[noun phrase] → [[determiner] [noun]]."

- **Discourse Indicators** are words that capture discourse relations among propositions, such as *because* and *therefore*. While most of them are meaningful in the cases where the propositions under consideration appear consecutively, a few of them are free from this restriction, as long as they share

the same topic. See Table 5.6 for the full list of discourse indicators.

Results. Table 5.2 summarises performances of each classifier on the test set under two different settings: *Scope* denotes whether all proposition pairs (*Global*) or only the pairs that are 2 propositions apart at most (*Local*) were used in the experiments. Both SVM and Naive Bayes classifiers were trained on the training set, whereas the rule-based classifier did not involve any training.

Table 5.2: Support vs no-relation classification results for ordered proposition pairs

Naive Bayes and SVM results are averages of 10 rounds of experiments with randomly downsampled training set to balance the class distribution. (For SVM, this approach led to better results than introducing class weights.)

Scope	Algorithm	Precision	Recall	F1	Accuracy
Global	Naive Bayes	0.02	0.94	0.05	0.16
	SVM	0.04	0.54	0.08	0.73
	Rule-based	1.00	0.42	**0.59**	**0.99**
Local	Naive Bayes	0.16	0.91	0.28	0.30
	SVM	0.24	0.49	**0.32**	**0.69**
	Rule-based	0.17	0.58	0.26	0.52

Both SVM and Naive Bayes perform poorly in the global scope, but much better in the local scope. While the global scope is a better representation of the real scenario, in which a given proposition can support any proposition in the thread, the class imbalance makes it a challenging learning problem. The negative instances, or ordered pairs in no-support relation, are more than 100 times the number of positive instances even after the preprocessing step. We tried to

remedy the problem by introducing class waits in SVM and downsampling the negative instances to balance the training set, but the approaches were not too effective.

Table 5.3 shows a clear difference in the set of most important features for SVM and Naive Bayes classifiers. Naive Bayes tends to put more weights to word pair features, whereas production rules are more important for SVM.

Table 5.3: Most Informative Features

Features listed in "+" and "-" rows are the most informative features associated with ordered proposition pairs in support and no support relation, respectively. Parenthesised features are word pair features, and features with arrows are production rules. Lastly, "[p]" means the given feature appears in the supporting proposition (premise), and "[c]" the supported proposition (conclusion).

Scope	Algorithm		Most Informative Features
Global	Naive Bayes	+	(i,be), (a,ban), (be,should), (only,the), (not,of)
		-	(be,do), whadjp → wrb jj [c], (?,?), (a,their), last token: '?' [p,c]
	SVM	+	s → np vp . [c], (you,you), vp → vbp adjp [p], s → np vp . [p]
		-	s → np vp [c], advp → rb [c], np → nn [c], np → nns [p]
Local	Naive Bayes	+	(flight,be), (the,must), (peanuts,peanuts), (peanuts,be)
		-	(to,just), frag → sbar . [c], (that,just), adjp → jj sbar [p]
	SVM	+	vp → vb adjp [p], root → s [c], (are,you), s → np advp vp . [c]
		-	np → nns [p], sbar → in s [c], np → prp [p], vp → vbp pp [p]

Word pairs "(peanuts, peanuts)" is correlated with support relation and "(?,?)" with no-support relation. The former suggests that propositions that share the same topic are more likely to be in support relation, and the latter shows that a question is unlikely to support another question. The most important feature for SVM in the local scope is having a verb phrase consisting of a

verb and adjective phrase in the supporting proposition. This could be hinting that supporting propositions often contain a detailed description.

The rule-based classifier[5], performs quite well in the global scope. A quick look at the confusion matrix (Table 5.4) reveals that this performance was made possible by the search space reduction step—all consecutive proposition pairs that are not in support relation were filtered out. We do not see the same effect in the local scope, however, resulting in a much lower precision.

Table 5.4: Confusion matrix for the classification results of the rule-based classifier (Global Scope)

		Predicted	
		Support	No-support
Actual	Support	161	116
	No-Support	0	12175

5.5 Taking an Interpretative Step

No matter how successful automatic mining of argument structure might be, the key challenge is then to provide information that allows sense to be made of the potentially very large datasets. A first example of such an interpretative step that offers end-users an insight into a debate is the notion of *centrality*. To pin down what we mean by centrality in this context, beyond mere intuition, we specify a calculation of an argument network structure. We construe that structure as a directed graph, $G = (V, E)$, in which vertices (V) are propositions

[5] As previously mentioned, the rule-based classifier simply returns "true" only when a given pair of propositions are consecutive and contain one or more discourse indicator.

or relations between propositions, and those relations are either support (pro arguments) or conflict (con arguments), captured by a function R which maps $V \mapsto \{prop, support, conflict\}$ and edges exist between them $E \subset V \times V$. Every relation may be further subtyped (i.e. classifying different types of support or conflict, etc.), but to keep the notation uncluttered we use a separate set of functions $R_{support}, R_{conflict}$ (abbreviated R_s and R_c) to encapsulate these taxonomies. For syntactic convenience, we refer to the number of edges at (i.e. the order of) a vertex v as $|v|$ and add a superscript to indicate whether we are interested in the number of incoming or outgoing edges, and a subscript to indicate constraints on the values of $R(v')$ of the vertex $v'$to which v is connected in each case. Thus, e.g. $|v|^{in}_{R(v')=support}$ is the number of edges incoming to v originating at vertices of type *support*. With this notation we can define centrality thus:

$$Central(v) = |v|^{in} + \sum_{v_i \text{s.t.}(v_i,v) \in E} Central(v_i) \tag{5.1}$$

That is, the centrality of a node v is simply the sum of the number of nodes rooted at v in the directed graph corresponding to the argument structure.

From the output of the classifier presented in Section 5.4, we are able to automatically generate argument maps corresponding to those in the manually annotated test corpus (an example is shown in Figure 5.1). We can then compare the calculated central issues for both manually and automatically annotated arguments. Those issues with a centrality score of 6 or greater are listed below:

Top central issues from the manually annotated corpus

1. but not restrictions on what people may choose to eat.
2. An outright ban should be in place.

3. The confined space and recycling of the air in a plane is a peanut allergy sufferers nightmare.
4. I am utterly amazed at the ignorance displayed by some of those commenting here.
5. but banning peanuts from flights via a DOT regulation seems to go too far.
6. treat your customers with disdain and make it as inconvenient as possible for them to use your product.
7. If the airline would normally have food for the passengers on that flight it seems silly to deny access to anything more than a bag of peanuts and a glass of water.

Top central issues from the automatic classification

1. An outright ban should be in place.
2. so perhaps those earlier rules should be revisited.
3. One item which should be addressed is food and drink.
4. but banning peanuts from flights via a DOT regulation seems to go too far.
5. but not restrictions on what people may choose to eat.
6. If the airline would normally have food for the passengers on that flight it seems silly to deny access to anything more than a bag of peanuts and a glass of water.
7. I am utterly amazed at the ignorance displayed by some of those commenting here.

Finally, we compare the complete ranking of issues by their centrality scores. To do this, we calculate two measures of distance between the rankings, the Euclidean Distance and the Manhattan Distance. The results as compared to a random baseline can be seen in Table 5.5.

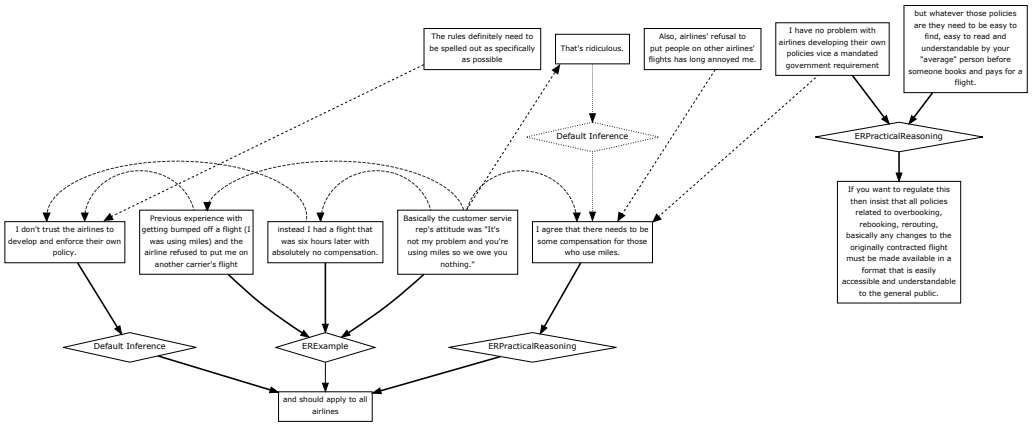

Figure 5.1: Argument map comparing manual and automatically identified connections. Correctly identified connections are in bold, false positives are dashed lines, and the single false negative is represented by dotted lines.

Table 5.5: The distance between the rankings of centrality scores for automatic annotation as compared to a random baseline

	Euclidean Distance	**Manhattan Distance**
Automatic Annotation	**3,845.08**	**76,696**
Random Baseline	5434.95	137,816

5.6 Conclusions

We have shown that, despite the challenges faced in understanding and summarising the large volumes of data that can be produced from online citizen dialogue, by analysing the argumentative structure contained within such a discussion we are able to obtain a deeper understanding of the issues being raised than by using existing techniques. Using the Argument Interchange Format to represent the argumentative structure, we are able to see not just points of agreement and disagreement, but to understand why those views are held and the expression of opinions both in support and in conflict with them.

We have highlighted several possible measures that can be determined from these structures, giving a clear insight into the topic and providing policy makers with tools to understand and interpret citizen dialogues. These include areas on which people generally agree and those areas which are central to the debate. We have selected a simple metric, centrality, to use as our exemplar and shown how even modest performance on the recovery of the argument network expressed in the discussion can yield robust results for this metric. The chapter has shown how a pipeline running through various computational linguistics techniques through analytical processes can be connected together. Though evaluation with users remains future work, the results in this chapter demonstrate for the first time that the state of the art in argument mining is already sufficient to start to offer real value to decision-makers and those responsible for public policy in interpreting and gaining insight into large-scale, complex debates.

In the following chapter, we present how a broader set of relations can be identified.

Table 5.6: Discourse indicators overview

From literature			
	List	**Indicates**	**Source - reference**
Inter	*because,therefore, after, for, since, when, assuming, so, accordingly, thus, hence, then, consequently*	Support	[42]
	however, but, though, except, not, never, no, whereas, nonetheless, yet,despite	Conflict	[42]
	as aresult	Conclusion	[86]
	reference to the first person in the covering sentence of an argument component: I,me, my, mine, and myself	Major claim	[73]
	while,whereas, whereas normally, whereas otherwise, not even, and yet	complementary coordinative argumentation	[81]
Intra	*cause, effect, means, end, makes that, leads to (and other expression which refer to causality only implicitly: cultivate, suddenly, in one blow, will yield, is, guarantee for, necessarily)*	causal argument	[81]
From eRulemaking training corpus			
	List	**Indicates**	**Source - corpus**
Intra	*argument*	indicates attacking	all attacking vs. all non-attacking
	you, your	weakly indicates attacking	all attacking vs. all non-attacking
	negative words ('funeral','death')	weakly indicates attacking	all attacking vs. all non-attacking
	should	strongly indicates supported	all supported vs. all non-supported
	I think	indicates supported	all supported vs. all non-supported
	you	strongly indicates ERAdhominem-attacking	all ERAdh-attacking vs. all non-ERAdh-attacking
	Personal pronouns (including possessive) , in order of keyness: him,his, he, our, my	strongly indicates ERExample-prem	all ERExample-prem vs. all non-ERExample-prem
	'association,(s)', 'cite', 'journal(s)', 'pages', 'published', 'studies', 'www', 'http', 'academy', 'college', 'reported', 'institute':	strongly indicates ERExpertOp-prem	all ERExpertOp-prem vs. all non-ERExpertOp-prem
	should	weakly indicates ERPractReas-concl	all ERPractReas-concl vs. all non-ERPractReas-concl

CHAPTER 6

IDENTIFYING A BROADER SET OF RELATIONS

In this chapter, we consider Step B in Figure 1.1 with a broader set of relations among propositions: *Support* is just one way in which propositions in a document can be related. There are many others, such as comparison and temporal.

We provide a systematic study of previously proposed features for *implicit discourse relation identification*, identifying new feature *combinations* that optimize F_1-score. The resulting classifiers achieve the best F_1-scores to date for the four top-level discourse relation classes of the Penn Discourse Tree Bank: COMPARISON, CONTINGENCY, EXPANSION, and TEMPORAL. We further identify factors for feature extraction that can have a major impact on the performance and determine that some features originally proposed for the task no longer provide performance gains in light of more powerful, recently discovered features. When originally published [59], our results constituted a new set of baselines for future studies of implicit discourse relation identification.

6.1 Background

The ability to recognize the discourse relations that exist between arbitrary text spans is crucial for understanding a given text. Indeed, a number of natural language processing (NLP) applications rely on it — e.g. question answering, text summarization, and textual entailment. Fortunately, *explicit discourse relations* — discourse relations marked by explicit connectives — have been shown to be easily identified by automatic means [66]: each such connective is gener-

ally strongly coupled with a particular relation. The connective "because", for example, serves as a prominent cue for the CONTINGENCY relation.

The identification of *implicit discourse relations* — where such connectives are absent — is much harder. It has been the subject of much recent research since the release of the Penn Discourse Treebank 2.0 (PDTB) [69], which annotates relations between adjacent text spans in Wall Street Journal (WSJ) articles, while clearly distinguishing *implicit* from *explicit* discourse relations.[1] Recent studies, for example, explored the utility of various classes of features for the task, including linguistically informed features, context, constituent and dependency parse features, and features that encode entity information or rely on language models [65, 45, 46, 94].

To date, however, there has not been a systematic study of combinations of these features for implicit discourse relation identification. In addition, the results of existing studies are often difficult to compare because of differences in data set creation, feature set choice, or experimental methodology. Hugo et al. [29], for example, explore the use of a new learning framework for the task (semi-supervised structural learning) using a single subset of known features rather than a subset chosen to duplicate previous studies or chosen on a per-relation basis.

This chapter provides a systematic study of previously proposed features for implicit discourse relation identification and identifies feature combinations that optimize F_1-score using greedy forward stepwise feature selection [15, 36]. We report the performance of our binary (one vs. rest) classifiers on the PDTB

[1] Research on implicit discourse relation recognition prior to the release of the PDTB instead relied on synthetic data created by removing explicit connectives from explicit discourse relation instances [48], but the trained classifiers do not perform as well on real-world data [10].

data set for its four top-level discourse relation classes: COMPARISON, CONTINGENCY, EXPANSION, and TEMPORAL. In each case, the resulting classifiers achieve the best F_1-scores for the PDTB to date. We further identify factors for feature extraction that can have a major impact on performance, including stemming and lexicon look-up. Finally, by documenting an easily replicable experimental methodology and making public the code for feature extraction[2], we hope to provide a new set of baselines for future studies of implicit discourse relation identification.

6.2 Data

The experiments are conducted on the PDTB [69], which provides discourse relation annotations between adjacent text spans in WSJ articles. Each training and test instance represents one such pair of text spans and is classified in the PDTB w.r.t. its **relation type** and **relation sense**.

In the work reported here, we use the **relation type** to distinguish examples of *explicit* vs. *implicit* discourse relations. In particular, we consider all instances with a relation type other than *explicit* as implicit relations since they lack an explicit connective between the text spans. The **relation sense** determines the relation that exists between its text span *arguments* as one of: COMPARISON, CONTINGENCY, EXPANSION, and TEMPORAL. For example, the following shows an explicit CONTINGENCY relation between *argument1* (arg1) and **argument2** (arg2), denoted via the *connective* "because":

The federal government suspended sales of U.S. savings bonds because **Congress**

[2]These are available from http://removed.for.anonymity.

hasn't listed the ceiling on government debt.

The four relation senses comprise the target classes for our classifiers.

A notable feature of the PDTB is that the annotation is done on the same corpus as Penn Treebank [49], which provides parse trees and part-of-speech (POS) tags. This enables the use of gold standard parse information for some features, e.g. the *production rules* feature, one of the most effective features proposed to date.

6.3 Features

Below are brief descriptions of features whose efficacy has been empirically determined in prior work[3], along with the rationales behind them:

Word Pairs (cross product of unigrams: arg1 × arg2) — A few of these word pairs may capture information revealing the discourse relation of the target spans. For instance, *rain-wet* can hint at CONTINGENCY.

First-Last-First3 (the first, last, and first three words of each argument) — The words in this range may be expressions that function as connectives for certain relations.

Polarity (the count of words in arg1 and arg2, respectively, that hold negated vs. non-negated positive, negative, and neutral sentiment) according to the MPQA corpus [90]) — The change in sentiment from arg1 to arg2 could be a good indi-

[3]Word Pairs [48]. First-Last-First3 [87]. Polarity, Verbs, Inquirer Tags, Modality, Context [65]. Production Rules [45].

cation of COMPARISON.

Inquirer Tags (negated and non-negated fine-grained semantic classification tags for the verbs in each argument and their cross product) — The tags are drawn from the General Inquirer Lexicon [75] which provides word level relations that might be propagated to the target spans' discourse relation, e.g. rise:fall.

Verbs (count of pairs of verbs from arg1 and arg2 belonging to the same Levin English Verb Class [54]; the average lengths of verb phrases as well as their cross product; and lastly, the POS of the main verb from each argument) — Levine Verb classes provide a means of clustering verbs according to their meanings and behaviors. Also, longer verb phrases might correlate with CONTINGENCY, indicating a justification.

Modality (three features denoting the presence of modal verbs in arg1, arg2, or both) — Modal verbs often appear in CONTINGENCY relations.

Context (the connective; the discourse relation senses for the immediately preceding and following relations if they are explicit relations; the location of the arguments within the paragraph) — Certain relations co-occur.

Production Rules (three features denoting the presence of syntactic productions in arg1, arg2 or both, based on all pairs of parent-children nodes in the argument parse trees) — The syntactic structure of an argument can influence that of the other argument as well as its relation type.

6.4 Experimental Setup

We aim to identify the optimal subsets of the aforementioned features for each of the four top-level PDTB discourse relation senses: COMPARISON, CONTINGENCY, EXPANSION, and TEMPORAL. In order to provide a meaningful comparison with existing work, we carefully follow the experiment setup of Pitler et al. [65], the origin of the majority of the features under consideration:

First, sections 0-2 and 21-22 of the PDTB are used as the validation and test set, respectively. Then, we randomly down-sample sections 2-20 to construct training sets for each of the classifiers, where each set has the same number of positive and negative instances with respect to the target relation. Since the composition of the corresponding training set has a noticeable impact on the classifier performances, we select a down-sampled training set for each classifier through cross validation. All instances of non-explicit relation senses are used; the ENTREL type is considered as having the EXPANSION sense.[4]

Second, Naive Bayes is used not only to duplicate the Pitler et al. [65] setting, but also because it equaled or outperformed other learning algorithms, such as SVM and MaxEnt, in preliminary experiments, while requiring a significantly shorter training time.[5]

Prior to the feature selection experiments, the best preprocessing methods for feature extraction are determined through cross validation. We consider simple lowercasing, Porter Stemming, PTB-style tokenization[6], and hand-crafted rules for matching tokens to entries in the polarity and General Inquirer lexi-

[4]Some prior work uses a different experimental setting. For instance, Zhou et al. [94] only considers two of the non-explicit relations, namely *Implicit* and *NoRel*.

[5]We use classifiers from the nltk package [8].

[6]Stanford Parser [38].

cons.

Then, feature selection is performed via forward stepwise regression, in which we start with the single best-performing feature and, in each iteration, add the feature that improves the F_1-score the most, until no significant improvement can be made. Once the optimal feature set for each relation sense is determined by testing on the validation set, we retrain each classifier using the entire training set and report final performance on the test set.

6.5 Results & Analysis

Feature Types	COMP. vs Rest		CONT. vs Rest		EXP. vs Rest		TEMP. vs Rest	
	F_1	Acc.	F_1	Acc.	F_1	Acc.	F_1	Acc.
1. Polarity	16.49	46.82	28.47	61.39	64.20	56.80	13.58	50.69
2. First-Last-First3	22.54	53.05	37.64	66.71	62.27	56.40	15.24	51.81
3. Inquirer Tags	18.07	82.14	34.88	69.60	77.76	66.38	21.65	80.04
4. Verbs	18.05	55.29	23.61	78.33	68.33	58.37	18.11	58.44
5. Production Rules	30.04	75.84	47.80	71.90	77.64	69.60	20.96	63.36
Best Combination	2 & 4 & 5		2 & 4 & 5		1 & 3 & 4 & 5		1 & 3 & 5	
	31.32	74.66	49.82	72.09	79.22	69.14	26.57	79.32
Pitler '09 (Best)	21.96	56.59	47.13	67.30	76.42	63.62	16.76	63.49
Zhou '10 (Best)[*]	*31.79*	*58.22*	*47.16*	*48.96*	*70.11*	*54.54*	*20.30*	*55.48*

[*] *The experiments are conducted under a slightly different setting, as described in Section 6.4.*

Table 6.1: Summary of Classifier Performances

Table 6.5 indicates the performance achieved by employing the feature set found to be optimal for each relation sense via forward stepwise regression, along with the performance of the individual features that constitute the ideal subset. The two bottom rows show the results reported in two previous papers with the most similar experiment methodology. The notable efficacy of *produc-*

tion rules feature, yiedling the best or the second best result across all relation senses w.r.t. both F_1-score and accuracy, confirms the finding of Zhou et al. [94]. In contrast to their work, however, combining existing features enhances the performance. Below, we discuss the primary observations gleaned from the experiments.

Word pairs as features. Starting with earlier works that proposed them as features [48], some form of *word pairs* has generally been part of feature sets for implicit discourse relation recognition. According to our research, however, these features provide little or no additional gain, once other features are employed. This seems sensible, since we now have a clearer idea of the types of information important for the task and have developed a variety of feature types, each of which aims to represent these specific aspects of the discourse relation arguments. Thus, general features like *word pairs* may no longer have a role to play for implicit discourse relation identification.

Preprocessing. Preprocessing turned out to impact the classifier performance immensely, especially for features like *polarity* and *inquirer tags* that rely on information retrieved from a lexicon. For these features, if a match for a given word is not found in the lexicon, no information is passed on to the classifier.

As an example, consider the General Inquirer lexicon. Most of its verb entries are present tense singular in form; thus, without stemming, dictionary look up fails for a large portion of the verbs. In our case, the F_1-score increases by roughly 10% after stemming.

Further tuning is possible by a few hand-written rules to guide lexicon lookup. The word *supplied*, for instance, becomes *suppli* after stemming, which

still fails to match the lexicon entry *supply*, unless adjusted accordingly.

Binning. An additional finding regards features that capture numeric, rather than binary, information, such as *polarity*. Since this feature encodes the counts of each type of sentiment words (with respect to each argument and their cross product), and Naive Bayes can only interpret binary features, we first employed a binning mechanism with each bin covering a single value. For instance, if arg1 consists of three positive words, we included *arg1pos1*, *arg1pos2* and *arg1pos3* as features instead of just *arg1pos3*.

The rationale behind binning is that it captures the proximity of related instances. Imagine having three instances each with one, two, and three positive words in arg1, respectively. Without binning, the features added are simply *arg1pos1*, *arg1pos2*, *arg1pos3*, respectively. From the perspective of the classifier, the third instance is no more similar to the second instance than it is to the first instance, even though having three positive words is clearly closer to having two positive words than having one positive word. With binning, this proximity is captured by the fact that the first instance has just one feature in common with the third instance, whereas the second instance has two.

Binning, however, results in single digit F_1-scores on most of the classification tasks. Without binning, the performance increases significantly. One possible explanation is that these features function as an abstraction of certain lexical patterns, rather than directly capturing similarities among the instances of the same class.

6.6 Conclusions

We explored a simple greedy feature selection approach to identify subsets of known features for implicit discourse relation identification that yield the best performance to date w.r.t. F_1-score on the PDTB data set. We also identified aspects of feature set extraction and representation that are crucial for obtaining state-of-the-art performance. Possible future work includes evaluating the performance without using the gold standard parses. This will give a better idea of how the features that rely on parser output will perform on real-world data where no gold standard parsing information is available. In this way, we can ensure that findings in this area of research bring practical gains to the community.

CHAPTER 7

CONCLUSIONS

Evaluating arguments has become an indispensable part of modern life. Our sources of information are no longer limited to books and articles produced by field experts and professional writers—With the advancement of information technology, the amount of textual content written by inexperienced writers, in the form of comments, product reviews, blog posts, etc., is steeply rising. The escalation of the amount of user generated text containing unclear argumentative structures and unsubstantiated claims is hampering effective communication among people. In commercial domains, this prevents individuals from conveying or acquiring useful information. In eRulemaking, this results in lost opportunities to share ideas and collectively solve critical problems the society is facing.

To remedy the issue, this thesis proposed automatic extraction and evaluation of argumentative structures in user comments in eRulemaking. More specifically, we leverage ideas from argumentation theory to formulate a model of argument to capture various types of propositions and their relations, synthesize NLP techniques to develop an argumentation mining tools to classify each argumentative proposition based on the type of appropriate support and recognize support relations among the propositions. The resulting abstraction of arguments can be evaluated via a quick comparison to the *evaluable* argument structure defined in the model. We have also constructed publicly available cor-

pora for the research community.

7.1 Future Work

Our vision is to facilitate discussion and debate among people through clear communication. There are many exciting opportunities that will collectively realize the vision by assisting people at both ends of communication. In this section, we discuss two applications of my dissertation work to help the writers and one to help the readers.

Machine-assisted Argument Construction: Amateur writers often make unsubstantiated claims. According to the human moderated online commenting project done at *Cornell eRulemaking Initiative (CeRI)*, people typically have reasons, and even objective evidence at times, supporting the claims they make. The problem is that they either do not realize the significance of providing support or simply forget to provide them. Through argument structure analysis, systems can identify unsupported claims and appropriate types of support for them. In online commenting setting, this can be implemented as part of the comment submission process—when a writer clicks the [submit] button, the system can notify the writer of unsupported statements to elicit reason or evidence, based on the type of the claim. This sort of feedback has an educational value as well, which is discussed below.

Writing Education: Education is essential for training individuals who can clearly communicate information and ideas through text. To make large scale

writing education and assessment feasible, there has been active research in automated essay grading for decades. However, state-of-the-art systems still use shallow NLP features like n-grams and sentence length. Incorporating argument structure analysis is likely to improve the performance of the systems, as it is one of the aspects mostly neglected by currently employed features. More importantly, this is a step beyond simply providing a *grade* to the students—feedback about argument structures, such as the statements that need further support, will better assist students in developing their writing skills.

Automated Document Summarization: Automatic summarization is a way to handle the overflow of textual content. Currently, active automatic summarization research is being conducted in various domains, such as legal, medical, and political documents. Modern summarization systems typically extract sentences from a given text to construct a summary, rather than generating their own sentences. Argument structure analysis can be incorporated into extractive summary approaches in several ways. One is to build a forest of argument trees. Collecting the root of each tree will result in a comprehensive extractive summary of the document. Better yet, we can build a *hierarchical* summary in which the user can expand the trees level by level for more details as necessary. Another potentially effective approach is to build the trees on the output of existing summarization systems. In that case, the goal would be to identify sentences that support other sentences in the summary. Such sentences can be excluded from the summary to produce a shorter summary with an equal or comparable coverage.

BIBLIOGRAPHY

[1] Kevin D. Ashley and Vern R. Walker. From information retrieval (ir) to argument retrieval (ar) for legal cases: Report on a baseline study. In *JURIX*, pages 29–38, 2013.

[2] Kedar Bellare, Gregory Druck, and Andrew McCallum. Alternating projections for learning with expectation constraints. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 43–50. AUAI Press, 2009.

[3] T. J. M. Bench-Capon. Deep models, normative reasoning and legal expert systems. In *Proceedings of the 2Nd International Conference on Artificial Intelligence and Law*, ICAIL '89, pages 37–45, New York, NY, USA, 1989. ACM.

[4] Jamal Bentahar, Bernard Moulin, and Micheline Bélanger. A taxonomy of argumentation models used for knowledge representation. *Artificial Intelligence Review*, 33(3):211–259, 2010.

[5] Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. An empirical investigation of statistical significance in nlp. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 995–1005, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[6] Philippe Besnard and Anthony Hunter. A logic-based theory of deductive arguments. *Artificial Intelligence*, 128(1):203 – 235, 2001.

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

[8] Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, COLING-ACL '06, pages 69–72, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[9] J. Anthony Blair. Walton's argumentation schemes for presumptive reasoning: A critique and development. *Argumentation*, 15(4):365–379, 2001.

[10] Sasha Blair-Goldensohn, Kathleen McKeown, and Owen Rambow. Building and refining rhetorical-semantic relation models. In *HLT-NAACL*, pages 428–435, 2007.

[11] Katarzyna Budzynska, Mathilde Janier, Juyeon Kang, Chris Reed, Patrick Saint-Dizier, Manfred Stede, and Olena Yaskorska. Towards argument mining from dialogue. In *Computational Models of Argument: Proceedings of COMMA 2014*, volume 266 of *Frontiers in Artificial Intelligence and Applications*, pages 185–196. IOS Press, 2014.

[12] Katarzyna Budzynska, Mathilde Janier, Chris Reed, and Patrick Saint-Dizier. Towards extraction of dialogical arguments. In *Proceedings of 13th International Conference on Computational Models of Natural Argument CMNA 2013*, 2013.

[13] Katarzyna Budzynska and Chris Reed. The structure of ad hominem dialogues. In *Proceedings of the 4th International Conference on Computational Models of Argument COMMA 2012*, pages 410–421, 2012.

[14] Elena Cabrio and Serena Villata. Combining textual entailment and argumentation theory for supporting online debates interactions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 208–212, Jeju Island, Korea, July 2012. Association for Computational Linguistics.

[15] Rich Caruana and Dayne Freitag. Greedy Attribute Selection. In W. Cohen and H. Hirsh, editors, *Proceedings of the Eleventh International Conference on Machine Learning*, pages 28–36. Morgan Kaufmann, 1994.

[16] Rowe Glenn Chris Reed, Raquel Mochales Palau and Marie-Francine Moens. Language resources for studying argument. In *Proceedings of the 6th conference on language resources and evaluation - LREC 2008*, pages 91–100. ELRA, 2008.

[17] Cary Coglianese. Citizen participation in rulemaking: past, present, and future. *Duke Law Journal*, pages 943–968, 2006.

[18] Koby Crammer and Yoram Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *J. Mach. Learn. Res.*, 2:265–292, March 2002.

[19] Hamish Cunningham, Diana Maynard, Kalina Bontcheva, Valentin Tablan, Niraj Aswani, Ian Roberts, Genevieve Gorrell, Adam Funk, Angus Roberts, Danica Damljanovic, Thomas Heitz, Mark A. Greenwood, Horacio Saggion, Johann Petrak, Yaoyong Li, and Wim Peters. *Text Processing with GATE (Version 6)*. 2011.

[20] Marie-Catherine De Marneffe, Bill Maccartney, and Christopher D. Manning. Generating typed dependency parses from phrase structure parses. In *In Proc. Int'l Conf. on Language Resources and Evaluation (LREC*, pages 449–454, 2006.

[21] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77(2):321 – 357, 1995.

[22] Bradley Efron and R.J. Tibshirani. *An Introduction to the Bootstrap*. Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis, 1994.

[23] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, June 2008.

[24] Cynthia R Farina, Mary Newhart, and Josiah Heidt. Rulemaking vs. democracy: Judging and nudging public participation that counts. *Mich. J. Envtl. & Admin. L.*, 2:123, 2012.

[25] Vanessa Wei Feng and Graeme Hirst. Classifying arguments by scheme. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 987–996, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[26] Kathleen Freeman. *Toward Formalizing Dialectical Argumentation*. PhD thesis, Eugene, OR, USA, 1993. UMI Order No. GAX94-05172.

[27] Kuzman Ganchev, Jennifer Gillenwater, and Ben Taskar. Dependency grammar induction via bitext projection constraints. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 369–377. Association for Computational Linguistics, 2009.

[28] Kuzman Ganchev, Joao Graça, Jennifer Gillenwater, and Ben Taskar. Posterior regularization for structured latent variable models. *The Journal of Machine Learning Research*, 99:2001–2049, 2010.

[29] Hugo Hernault, Danushka Bollegala, and Mitsuru Ishizuka. Semi-supervised discourse relation classification with structural learning. In *CICLing (1)*, pages 340–352, 2011.

[30] David Hitchcock. Toulmin's warrants. In FransH. Van Eemeren, J.Anthony Blair, CharlesA. Willard, and A.Francisca Snoeck Henkemans, editors, *Anyone Who Has a View*, volume 8 of *Argumentation Library*, pages 69–82. Springer Netherlands, 2003.

[31] David Hitchcock. Good reasoning on the toulmin model. *Argumentation*, 19(3):373–391, 2005.

[32] T. A. Hollihan and K. T. Baaske. *Arguments and Arguing: The Products and Process of Human Decision Making, Second Edition*. Waveland Press, 2004.

[33] Mark Howard. E-government across the globe: how will'e'change government. *e-Government*, 90:80, 2001.

[34] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Trans. Neur. Netw.*, 13(2):415–425, March 2002.

[35] Mathilde Janier, John Lawrence, and Chris Reed. Ova+: an argument analysis interface. In *Proceedings of the Fifth International Conference on Computational Models of Argument (COMMA 2014)*, pages 463–464, 2014.

[36] George H. John, Ron Kohavi, and Karl Pfleger. Irrelevant features and the subset selection problem. In W. Cohen and H. Hirsh, editors, *Proceedings of the Eleventh International Conference on Machine Learning*, pages 121–129. Morgan Kaufmann, 1994.

[37] S. Sathiya Keerthi, S. Sundararajan, Kai-Wei Chang, Cho-Jui Hsieh, and Chih-Jen Lin. A sequential dual method for large scale multi-class linear svms. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '08, pages 408–416, New York, NY, USA, 2008. ACM.

[38] Dan Klein and Christopher D. Manning. Accurate unlexicalized parsing. In *IN PROCEEDINGS OF THE 41ST ANNUAL MEETING OF THE ASSOCIATION FOR COMPUTATIONAL LINGUISTICS*, pages 423–430, 2003.

[39] Barbara Konat, John Lawrence, Joonsuk Park, Katarzyna Budzynska, and Chris Reed. A corpus of argument networks: Using graph properties to analyse divisive issues. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016.

[40] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Con-

ditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, 2001.

[41] John Lawrence, Floris Bex, Chris Reed, and Mark Snaith. Aifdb: Infrastructure for the argument web. In *Proceedings of the 4th International Conference on Computational Models of Argument COMMA*, pages 515–516, 2012.

[42] John Lawrence and Chris Reed. Combining argument mining techniques. In *Working Notes of the 2nd Argumentation Mining Workshop, ACL'2015*, 2015.

[43] John Lawrence, Chris Reed, Colin Allen, Simon McAlister, and Andrew Ravenscroft. Mining arguments from 19th century philosophical texts using topic based modelling. In *Proceedings of the First Workshop on Argumentation Mining*, pages 79–87, Baltimore, Maryland, June 2014. Association for Computational Linguistics.

[44] Quoc V Le and Tomas Mikolov. Distributed representations of sentences and documents. In *ICML*, volume 14, pages 1188–1196, 2014.

[45] Ziheng Lin, Min-Yen Kan, and Hwee Tou Ng. Recognizing implicit discourse relations in the penn discourse treebank. In *EMNLP*, pages 343–351, 2009.

[46] Annie Louis, Aravind K. Joshi, Rashmi Prasad, and Ani Nenkova. Using entity features to classify implicit discourse relations. In *SIGDIAL Conference*, pages 59–62, 2010.

[47] J.S. Lubbers, American Bar Association. Section of Administrative Law, Regulatory Practice, American Bar Association. Government, and Public Sector Lawyers Division. *A Guide to Federal Agency Rulemaking*. ABA Section of Administrative Law and Regulatory Practice and Government and Public Sector Lawyers Division, 2012.

[48] Daniel Marcu and Abdessamad Echihabi. An unsupervised approach to recognizing discourse relations. In *ACL*, pages 368–375, 2002.

[49] Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *COMPUTATIONAL LINGUISTICS*, 19(2):313–330, 1993.

[50] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean.

Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[51] Marie-Francine Moens, Erik Boiy, Raquel Mochales Palau, and Chris Reed. Automatic detection of arguments in legal texts. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law*, ICAIL '07, pages 225–230, New York, NY, USA, 2007. ACM.

[52] Marie-Francine Moens, Erik Boiy, Raquel Mochales Palau, and Chris Reed. Automatic detection of arguments in legal texts. In *Proceedings of the 11th international conference on Artificial intelligence and law*, pages 225–230. ACM, 2007.

[53] M. Jae Moon. The evolution of e-government among municipalities: rhetoric or reality? *Public administration review*, 62(4):424–433, 2002.

[54] Beth Levin Northwestern, Beth Levin, and Harold Somers. English verb classes and alternations: A preliminary investigation, 1993.

[55] Tim O'Keefe and Irena Koprinska. Feature selection and weighting methods in sentiment analysis. In *Proceedings of the 14th Australasian Document Computing Symposium*, 2009.

[56] Raquel Mochales Palau and Marie-Francine Moens. Argumentation mining: The detection, classification and structure of arguments in text. In *Proceedings of the 12th International Conference on Artificial Intelligence and Law*, ICAIL '09, pages 98–107, New York, NY, USA, 2009. ACM.

[57] Raquel Mochales Palau and Marie-Francine Moens. Argumentation mining: the detection, classification and structure of arguments in text. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 98–107. ACM, 2009.

[58] Joonsuk Park, Cheryle Blake, and Claire Cardie. Toward machine-assisted participation in erulemaking: An argumentation model of evaluability. *Proceedings of the Fifteenth International Conference on Artificial Intelligence and Law (ICAIL)*, 2015.

[59] Joonsuk Park and Claire Cardie. Improving implicit discourse relation recognition through feature set optimization. In *Proceedings of the 13th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, SIGDIAL '12, pages 108–112, Stroudsburg, PA, USA, 2012. ACL.

[60] Joonsuk Park and Claire Cardie. Identifying appropriate support for propositions in online user comments. In *Proceedings of the First Workshop on Argumentation Mining*, pages 29–38, Baltimore, Maryland, June 2014. ACL.

[61] Joonsuk Park, Arzoo Katiyar, and Bishan Yang. Conditional random fields for identifying appropriate types of support for propositions in online user comments. In *Proceedings of the 2nd Workshop on Argumentation Mining*, pages 39–44, Denver, CO, June 2015. ACL.

[62] Joonsuk Park, Sally Klingel, Claire Cardie, Mary Newhart, Cynthia Farina, and Joan-Josep Vallbé. Facilitative moderation for online participation in erulemaking. In *Proceedings of the 13th Annual International Conference on Digital Government Research*, pages 173–182. ACM, 2012.

[63] Andreas Peldszus. Towards segment-based recognition of argumentation structure in short texts. In *Proceedings of First Workshop on Argumentation Mining*, 2014.

[64] Andreas Peldszus and Manfred Stede. Towards detecting counter-considerations in text. In *2nd Workshop on Argumentation Mining (ARG-MINING 2015)*, 2015.

[65] Emily Pitler, Annie Louis, and Ani Nenkova. Automatic sense prediction for implicit discourse relations in text. In *ACL/AFNLP*, pages 683–691, 2009.

[66] Emily Pitler, Mridhula Raghupathy, Hena Mehta, Ani Nenkovak, Alan Lee, and Aravind K. Joshi. Easily identifiable discourse relations. In *COLING (Posters)*, pages 87–90, 2008.

[67] John L. Pollock. Defeasible reasoning. *Cognitive Science*, 11:481–518, 1987.

[68] Henry Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2011.

[69] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. The penn discourse treebank 2.0. In *Proceedings of LREC*, 2008.

[70] Iyad Rahwan, Fouad Zablith, and Chris Reed. Laying the foundations for a worldwide argument web. *Artificial Intelligence*, 171(10-15):897–921, 2007.

[71] Paul Rayson, Andrew Wilson, and Geoffrey Leech. Grammatical word

class variation within the british national corpus sampler, 2001. Language and Computers.

[72] Ellen Riloff and Jay Shoen. Automatically acquiring conceptual patterns without an annotated corpus. In *In Proceedings of the Third Workshop on Very Large Corpora*, pages 148–161, 1995.

[73] Christian Stab and Iryna Gurevych. Annotating argument components and relations in persuasive essays. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 1501–1510, 2014.

[74] Gries Stefan. Quantitative corpus linguistics with r. a practical introduction, 2009.

[75] Philip J. Stone, Dexter C. Dunphy, Marshall S. Smith, and Daniel M. Ogilvie. *The General Inquirer: A Computer Approach to Content Analysis*, volume 08. MIT Press, 1966.

[76] Simone Teufel. *Argumentative Zoning: Information Extraction from Scientific Text*. PhD thesis, University of Edinburgh, 1999.

[77] Simone Teufel and Marc Moens. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational linguistics*, 28(4):409–445, 2002.

[78] Stephen E Toulmin. *The Uses of Argument*. Cambridge University Press, 1958.

[79] Stephen E. Toulmin, Richard Rieke, and Allan Janik. *An Introduction to Reasoning*. Macmillan Publishing Company, 1979.

[80] Frans H. van Eemeren, Rob Grootendorst, and Tjark Kruiger. *Handbook of argumentation theory ; a critical survey of classical backgrounds and modern studies*. PDA Series. Foris Publications, 1987.

[81] Frans H. van Eemeren, Peter Houtlosser, and Arnolda Francisca Snoeck Henkemans. *Argumentative indicators in discourse: A pragma-dialectical study*, volume 12. Springer Science & Business Media, 2007.

[82] Maria Paz Garcia Villalba and Patrick Saint-Dizier. Some facets of argument mining for opinion analysis. In *COMMA*, pages 23–34, 2012.

[83] Douglas Walton. *Argumentation schemes for presumptive reasoning*. Lawrence Erlbaum Associates, 1996.

[84] Douglas Walton. *Appeal to Expert Opinion*. Penn State Press, Univesity Press, USA, 1997.

[85] Douglas Walton and Giovanni Sartor. Teleological justification of argumentation schemes. *Argumentation*, 27(2):111–142, 2013.

[86] Bonnie Webber, Markus Egg, and Valia Kordoni. Discourse structure and language technology. *Natural Language Engineering*, 18(04):437–490, 2012.

[87] Ben Wellner, Lisa Ferro, Warren R. Greiff, and Lynette Hirschman. Reading comprehension tests for computer-based understanding evaluation. *Natural Language Engineering*, 12(4):305–334, 2006.

[88] Theresa Wilson. Recognizing contextual polarity in phrase-level sentiment analysis. In *In Proceedings of HLT-EMNLP*, pages 347–354, 2005.

[89] Theresa Wilson and Janyce Wiebe. Annotating attributions and private states. In *Proceedings of ACL Workshop on Frontiers in Corpus Annotation II: Pie in the Sky*, 2005.

[90] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *HLT/EMNLP*, 2005.

[91] N.V. Wood. *Perspectives on Argument*. Pearson/Prentice Hall, 2006.

[92] Adam Wyner, Raquel Mochales-Palau, Marie-Francine Moens, and David Milward. Semantic processing of legal texts. chapter Approaches to Text Mining Arguments from Legal Cases, pages 60–79. Springer-Verlag, Berlin, Heidelberg, 2010.

[93] Bishan Yang and Claire Cardie. Context-aware learning for sentence-level sentiment analysis with posterior regularization. In *Proceedings of ACL*, 2014.

[94] Zhi-Min Zhou, Yu Xu, Zheng-Yu Niu, Man Lan, Jian Su, and Chew Lim Tan. Predicting discourse connectives for implicit discourse relation recognition. In *COLING (Posters)*, pages 1507–1514, 2010.